

Dr. ZÁTONYI SÁNDOR /  
ZÁTONYI SÁNDOR

# C 16, Plus / 4

az általános iskolai  
fizikatanításban

**NOVOTRADER**



Dr. Zátanyi Sándor -- Zátanyi Sándor:

**C-16, Plus/4 az általános iskolai  
fizikatanításban**

1987. NOVOTRADE RT. DELTASoft Iroda

Lektorálta: Kiss Erzsébet szaktanár  
Mása István programozó

A kiadásért felel: Rényi Gábor, a NOVOTRADE Rt. igazgatója

ISBN 963 02 5105 1

Copyright Dr. Zátonyi Sándor - Zátonyi Sándor

Készült a Badacsonyi Állami Gazdaság nyomdaüzemében

Felelős vezető: Töttössy Péter

# 1. A SZÁMÍTÓGÉP ALKALMAZÁSÁNAK METODIKÁJA

A számítógép iskolai használatának alapvetően két módja van: lehet a számítógép oktatási eszköz, amellyel segíthetjük az egyes tantárgyak anyagának feldolgozását, és lehet a számítógép tananyag azzal a céllal, hogy megismerjék a tanulók a programozást, a számítógép felhasználását.

Könyvünk a számítógép oktatási eszközként történő alkalmazásához kíván segítséget nyújtani. Pontosabban: a Commodore 16-os számítógépnek az általános iskolai fizikatanításban, -tanulásban való alkalmazásához ad módszertani javaslatokat, kész programokat. Az általános iskolákban ma a C-16-os számítógép az egyik legelterjedtebb típus, célszerű volt tehát ehhez igazodva kidolgozni ajánlásainkat. A közölt programok azonban minden módosítás nélkül alkalmazhatók a Commodore Plus/4 típusú számítógépen is. (A közeljövőben számítani lehet e gépek nagyobb arányú elterjedésére is.)

A könyvben található programok felhasználásához csak a C-16-os (C Plus/4-es) számítógép kezelésének ismeretére van szükség. Az alkalmazás egyes lépéseihez a programok bevezető szövegében, illetve a képernyőn megjelenő tájékoztatóban olvashatunk konkrét útmutatást.

## 1.1. A számítógép alkalmazása a tanítási órán és a tanórán kívüli foglalkozásokon

A számítógép egyaránt eredményesen alkalmazható a tanítási órán és a tanulók különböző csoportjai számára szervezett foglalkozásokon: a szakkörökön, a fakultatív foglalkozáson, a korrepetáción. Ezek részben azonos, részben eltérő tartalmi és szervezési kérdéseket vetnek fel. A következőkben ezeket tekintjük át.

### 1.1.1. Számítógép a tanítási órán

A fizikaórákon elsődlegesen olyan témájú programokat célszerű alkalmaznunk, amelyek közvetlenül segítik a tanterv törzsanyagának megértését, elsajátítását, alkalmazását, rögzítését, ellenőrzését. A tananyagot olyan szinten kell alkalmaznia a programnak, amely jól követhető a tanulók többsége számára. Különösen előnyös az olyan program alkalmazása, amely lehetővé teszi ugyanannak a jelenségnek, összefüggésnek különböző szinten történő feldolgozását.

A tanítási órán a tananyagot általában osztálykeretben ("frontális" munkában) dolgozzuk fel. Ezért a tanulólétszámtól függően 3-5 monitort (TV-készüléket) kell beállítanunk és egyidejűleg működtetnünk ugyanazzal a számítógéppel, hogy valamennyi tanuló jól lássa a képernyőt.

Az ismeretek alkalmazásának fázisában esetenként csoportmunkában is foglalkoztatjuk az osztály tanulóit, vagy a tanulók egy részét (differenciált munka).

Ebben az esetben külön-külön programról kell gondoskodnunk minden csoport számára. Az ismétlés időszakában eltérő témájúak is lehetnek ezek a programok. Ügyelnünk kell azonban arra, hogy a program szintje alkalmazkodjék az adott csoport korábbi eredményeihez, s az fejlesztő hatású legyen számukra.

Csoportmunka esetén külön-külön számítógépet és monitort szükséges biztosítanunk mindegyik csoport számára, s úgy kell elhelyeznünk a csoportokat a teremben, hogy azok ne zavarják egymás munkáját.

A témakörök összefoglalása után és az év végi ismétlés időszakában olyan szervezeti megoldást is választhatunk, hogy egy-egy tanulónak **egyéni munkát** adunk a számítógép segítségével, a többi tanulóval pedig közvetlenül foglalkozunk. Ilyen egyéni munkát adhatunk olyan tanulónak, aki biztonságosan tudja azt a tananyagot, amivel az osztály foglalkozik, s számára magasabb teljesítményt igénylő feladatot tűzünk ki. Indokolhatja az egyéni munkát az is, ha a tanuló osztálytársaitól lemaradt, s egy meghatározott témában gyakorlásra van szüksége a felzárkóztatáshoz, az alapvető ismeretek pótlásához.

Ilyen szervezeti megoldás esetén természetesen külön-külön számítógépet és monitort kell biztosítanunk minden, egyéni munkát végző tanuló számára. Azt javasoljuk, hogy maximálisan három-négy tanuló számára adjunk egyidejűleg egyéni munkát, mivel több tanuló esetén túl sok időt vesz igénybe a program betöltése, s nehéz - a többi tanulóval való foglalkozás mellett - a munkájuk áttekintése, óra végi értékelése.

### 1.1.2. Számítógép a fakultatív foglalkozáson és a szakkörön

A fakultatív foglalkozás és a szakkör közös sajátossága, hogy azon elsődlegesen azok a tanulók vesznek részt, akik **kiemelkedő képességűek**, jó eredményt értek el fizikából, de legalább is érdeklődést mutatnak a tantárgy iránt. Ebből adódik, hogy a fakultatív foglalkozáson, szakkörön olyan ismeretek feldolgozását tervezhetjük, amelyek **túlmutatnak a tanítási órán feldolgozott anyagon**. Ez nemcsak lehetőség, hanem szükségszerű is ahhoz, hogy a foglalkozás ténylegesen bővítse a tanulók ismereteit, fejlessze képességeiket.

A fakultatív foglalkozáson és a szakkörön ezért olyan programok alkalmazását javasoljuk, amelyek a tanórainál **magasabb követelmények** elé állítják a tanulókat. Megkönnyíti a számítógép alkalmazását az a tény, hogy e foglalkozásokon jelentősen kisebb a tanulók létszáma, mint a tanítási órán.

### 1.1.3. Korrepetálás számítógéppel

A tanulók különböző hatásfokkal tudják feldolgozni a tantervi anyagot, s vannak tanulók, akik nem érik el a **továbbhaladáshoz szükséges minimum szintet** sem. Az ő számukra olyan egyéni (vagy néhány tanulóból álló kiscsoportos) tanórán kívüli foglalkozást szükséges szerveznünk, amely lehetőséget biztosít számukra a felzárkózáshoz, vagy legalább a minimum szint eléréséhez.

Korrepetálást olyan tanulók számára indokolt szervezni, akik **önhibájukon kívül** (hosszabb hiányzás, lassú felfogóképesség, vagy más ok miatt) nyújtanak gyengébb teljesítményt. A jó képességű, de hanyag, lemaradó tanulók problémájára nem a korrepetálás ad megoldást.

A korrepetáláson olyan programok alkalmazása célszerű, amelyek elsődlegesen a **minimum szintű követelmények teljesítéséhez, az alapképességek fejlesztéséhez** adnak segítséget. Fizikából különösen az egyszerű számításos feladatok megoldásához adhatnak jelentős segítséget az olyan programok, amelyek különböző nehézségű, szöveg nélküli és szöveges feladatokat tartalmaznak.

E programok többsége olyan értékelő részt is tartalmaz, amely százalékosan megadja, hogy a tanuló milyen eredményt ért el a különböző szintű feladatok megoldásában. Azt javasoljuk, hogy ezt a részt a tanulók önértékeléséhez hasznosítsuk, s **ne ez alapján adjunk érdemjegyet** a tanulók munkájára.

A korrepetáláshoz természetesen minden tanuló számára külön számítógépet kell biztosítanunk, s azt a tanulóknak maguknak kell kezelniük. Miután betöltöttük a programot, a tanulóknak szükséges megadnunk a gép kezeléséhez szükséges legfontosabb tájékoztatást.

Első lépésként csak annyi információt célszerű adni, amennyi az adott program használatához szükséges. Tapasztalataink szerint ehhez néhány perc azon tanulók számára is elegendő, akik korábban nem használtak számítógépet.

- x -

A könyvünkben közölt programok mindegyike **közvetlenül kapcsolódik az általános iskolai fizika tantervi anyagához**. Azt, hogy e programok közül melyiket tudjuk az adott osztályban a tanítási órán, vagy órán kívüli foglalkozáson felhasználni, az a program tartalmán kívül jelentős mértékben függ az osztály fejlettségi szintjétől és a soron következő óra, foglalkozás didaktikai feladatától.

## **1.2. A számítógép alkalmazása és a hagyományos módszerek kapcsolata**

Az általános iskolai fizikatanítás, -tanulás folyamatában kiemelkedő jelentősége van a **tanulói kísérletezésnek, mérésnek**. A tanulók kísérletezés, mérés közben aktív részesei a fizikai jelenségek létrehozásának, a változások előidézésének, megfigyelésének, a tapasztalatok lejegyzésének, elemzésének, s a következtetések levonásának. A rendszeresen végzett tanulói kísérletek, mérések meggyőzőbbek, a szerzett ismeretek tartósabbak, mint ha csak szemlélői a tanulók a kísérletek elvégzésének. A tanulói kísérletek, mérések pozitív hatással vannak a fizika iránti érdeklődés, a motiváció alakulására is.

A tanulói kísérlet és mérés mellett fontos szerepe van a fizikatanításban a **tanári kísérletnek, mérésnek** is. Tanári kísérletek, mérések végzése célszerű minden olyan esetben, amikor túlzottan összetett a kísérleti berendezés, vagy kezelése nagy gyakorlottságot igényel, amikor a tanulói kísérlet előkészítésére, végzésére fordított energia és idő nem állna arányban az elérhető eredménnyel, amikor nem áll rendelkezésünkre kellő számú tanulókísérleti eszköz. Fontosnak tartjuk, hogy a tanári kísérletek, mérések bemutatása után is a tanulók önállóan végezzék

a kísérleti tapasztalatok, mérések eredményeinek a lejegyzését, elemzését, a következtetések levonását.

A számítógépet a következő módon alkalmazhatjuk a fizikai jelenségek bemutatására, illetve a mérési eredmények pontosabbá tételéhez, a tapasztalatok elemzéséhez:

#### *a/ A számítógép mint kísérleti eszköz*

Egyes esetekben a számítógép önmagában is alkalmas arra, hogy fizikai jelenséget mutassunk be. A rezgőmozgás tanításakor például bemutathatjuk segítségével, hogy milyen összefüggés van a rezgésszám és a hang magassága között. Miközben a képernyőről egyre nagyobb rezgésszámot olvashatunk le, a hangszóróból egyre magasabb hangot hallunk.

Jól érzékeltethető a számítógéppel az interferencia jelensége is. Két különböző rezgésszámú hangot szólaltatunk meg, s változtatjuk az egyik rezgésszámát. Megfelelő rezgésszám-különbség esetén jól hallani az egyenlő időközökben bekövetkező hangerősödést, -gyengülést.

#### *b/ A mérési eredmények pontosabbá tétele*

A számítógépbe beépített óra és az elektromos csatlakozási lehetőség - megfelelő program alkalmazása esetén - lehetővé teszi a kísérletek, mérések pontosabb végzését. A szabadesés jelenségének vizsgálatakor például megoldható, hogy a számítógép segítségével századmásodperc pontossággal megmérjük az acélgolyó elengedésétől a földre esésig eltelt időt.

Számos további lehetőség adódik az egyenletes és a változó mozgás kísérleti vizsgálatára a számítógép segítségével. E vizsgálatok, mérések széles körű megvalósítására azonban csak akkor kerülhet sor, ha központilag sikerül biztosítani az ehhez szükséges kiegészítő eszközöket.

#### *c/ A jelenségek szimulálása a számítógéppel*

Miután kísérletekre alapozottan megismerték a tanulók az egyes fizikai jelenségeket, összefüggéseket, a számítógép lehetőséget nyújt arra, hogy a jelenség lefolyását a tanult összefüggéseknek megfelelő módon a képernyőn sematikus, mozgó ábrák segítségével megjelenítsük, megfigyeltezzük. Óriási előnye e módszernek, hogy a jelenség azonos módon rövid idő alatt ismételhető, egyes mozzanatok "kimerevíthetők". Mi magunk, vagy a tanulók adhatják meg az egyes mennyiségeket, így a különböző tényezőktől való függés könnyen, jól szemléltethető. Az is megoldható, hogy a számítógép folyamatosan jelezze a változó mennyiségeket, s megadja a "betáplált" adatok alapján a végső eredményt. Különösen szemléletes a tanulók számára az, ha a jelenség szimulálása mellett a gép grafikonon is rögzíti a mennyiségek közti összefüggést.

Jól, szemléletesen szimulálható a számítógéppel például a lencsék képalkotása, a testek ütközése. A jelenség érzékeltetése mellett a grafikon folyamatos rajzolásával is segíthetjük a megértést, például az egyenesvonalú egyenletes és változó mozgás, a rezgőmozgás, Ohm törvénye, a termikus kölcsönhatás tanításakor.



A számítógép tehát sokoldalú kiegészítést, megerősítést nyújt a kísérletezéshez, segíti a kísérleti tapasztalatok, mérési eredmények elemzését, a tanult összefüggések megerősítését. E sok pozitívum ellenére sem helyettesítheti, pótolhatja a számítógép alkalmazása a kísérletek elvégzését, a jelenségek tényleges bemutatását. A kettő kölcsönösen kiegészíti egymást.

Az audiovizuális eszközök közül az írásvetítő a legelterjedtebb az általános iskolai fizikatanításban. Az írásvetítő elsősorban olyan jelenségek, összefüggések bemutatására alkalmas, amelyek megértéséhez fokozatos kialakítás, kiegészítés szükséges, és viszonylag hosszabb időn át kell a kivetített szöveget, képet szemlélni. Jól alkalmazhatók a többrétegű, mozgatható részeket is tartalmazó transzparenszek. Nagy előnye az írásvetítőnek a hagyományos tábla alkalmazásával szemben, hogy előre felírható a fóliára a szöveg, elkészíthető a rajz, s így nem kell azzal időt tölteni a tanítási órán.

A számítógép annyiban ad újat, többet az írásvetítőhöz viszonyítva, hogy a gép memóriájában (vagy a magnószalagon) sokféle összefüggés, adat tárolható, s azok változatos formában hasznosíthatók a tanult ismeretek megerősítésére, gyakorlására.

A különböző mennyiségek közötti összefüggések szemléltetésére például olyan táblázatokat közölhetünk a számítógép segítségével, amelyeken egy-egy adatsor megváltoztatásával nyomon követhetjük, hogy milyen módon változik meg a másik (többi) mennyiség. A különböző anyagú, egyenlő térfogatú testek tömegének összehasonlítása után néhány másodperc múlva már az egyenlő tömegű testek térfogatát tudjuk összehasonlítani.

Gyorsabban, egyszerűbben rajzoltathatjuk meg a grafikonokat is a számítógéppel, mint az írásvetítővel. A számítógép ezen túl még azt is lehetővé teszi, hogy az órán újólág felmerülő igényeknek megfelelően rajzoltassunk fel új és új grafikonokat, összehasonlítva azok jellemzőit.

Mindezek ellenére a számítógép nem teszi feleslegessé az írásvetítő alkalmazását sem, mivel a falra vetíthető nagy kép, az egyedi megoldások alkalmazása, a vetítési lehetőség egyszerűsége másfajta előnyöket biztosít, mint a számítógép.

A többi audiovizuális eszköz (diavetítő, filmvetítő, televízió, képmagnó) alkalmazását elsősorban olyan szempontból tartjuk nélkülözhetetlen módszernek, hogy segítségükkel átmenetet képezhetünk a valóság tárgyai, jelenségei és a tanulók tudatában kialakuló képzetek, fogalmak, ítéletek között. A tanulók jól megfigyelhetik például a filmvetítés vagy a televízióadás során, hogy mi történik akkor, ha egy mozgó vasúti kocsinál egy álló vasúti kocsinak ütközik. A képi megjelenítés jól tükrözi a valóságot, de már egy lépés az absztrakció felé. A számítógéppel végzett szimuláció már ehhez képest is egy további lépés, mivel a képernyőn már nem a tárgyak tényleges képét, hanem azok szimbolizált rajzát látják a tanulók.

Meghatározó szerepe van a fizikatanításban a tanári magyarázatnak és a beszélgetés módszerének. A tankönyv a tantervi anyag feldolgozását meghatározott logikai felépítésben tartalmazza. Az egyes logikai lépések részletes kifejtése, magyarázata azonban nem léphet túl egy optimális mértéket, mert ez már veszélyeztetné a fő gondolatmenet követhetőségét. Ugyanakkor a tankönyvi feldolgozás részletezettsége a tanulócsoporthoz egy részének jelent a

felkészültségének megfelelő "bontottságot". A magyarázattal, illetve a kérdések, válaszok révén folytatott beszélgetéssel kell biztosítanunk a tanulók számára a tankönyvnél részletesebb tananyag-kifejtést. A beszélgetés módszerének alkalmazása során ítélni kell meg azt, hogy mely részeknél szükséges részletesebb vagy kevésbé részletes magyarázat. A tanulók visszajelzéseire figyelve, azokhoz igazodva kell biztosítanunk az új ismeretek korábbiakhoz való kapcsolódását, az egyes tények, adatok közötti összefüggések felismerését, az általánosításhoz való eljutást, a megismert törvények, összefüggések alkalmazását.

**A számítógép jó kiegészítést adhat a tanári magyarázat és a beszélgetés módszeréhez.** A termikus kölcsönhatás magyarázatához például nagyon szemléletes képet adhatunk a számítógéppel annak érzékeltetéséhez, hogy miként befolyásolja a közös hőmérséklet kialakulását a két test tömege, vagy a két test anyagának fajhője (minden más változatlanul hagyása mellett).

Jól szemléltethető az ellenállás-adatok értelmezése is a számítógépes programmal. Konkrét adatpárok alapján figyelhetik meg a tanulók, hogy ahányszorosára nő a fogyasztó két kivezetése között mért feszültség, ugyanannyiszorosára nő a fogyasztón áthaladó áram erőssége is. Közben a kettő hányadosa állandó.

A fizikai ismeretek alkalmazásának, megszilárdításának hatékony módszere a **feladatmegoldás**. Amikor feladatot oldatunk meg a tanulókkal, tulajdonképpen azt várjuk tőlük, hogy a feladat konkrét tartalmából, adataiból kiindulva keressék azt a tanult összefüggést, törvényt, amely az adott témával kapcsolatos, s azt átalakítva, aktualizálva adjanak választ a feladat kérdésére.

Általános tapasztalat, hogy a tanulók számára különösen a számításos feladatok megoldása jelent nehézséget. **A feladatmegoldások gyakorlásában jól hasznosíthatjuk a számítógépadta lehetőségeket.** A tanulókkal - differenciált foglalkoztatás vagy egyéni megbízás keretében - olyan témájú feladatokat oldathatunk meg, amelyekben szükségesnek látjuk a fejlesztést, a feladatok nehézségi szintje pedig a tanulók korábbi teljesítményéhez igazodhat.

### 1.3. Módszertani javaslatok a programok felhasználásához

A könyvben közölt programok egy része elsősorban az új ismeretek feldolgozásához, más része főleg az alkalmazáshoz, illetve az ellenőrzéshez használható. Vannak olyan programok is, amelyek egyaránt alkalmazhatók a tanítási-tanulási folyamat mindegyik fázisában. A felhasználás módja, a tanulók aktivizálásának lehetséges mértéke, szintje azonban más-más akkor, ha ugyanazt a programot az új anyag feldolgozásához, vagy a gyakorláshoz, illetve az ellenőrzéshez alkalmazzuk.

A következőkben néhány módszertani javaslatot szeretnénk adni a programok felhasználásához. Javaslatainkat a tanítási órák és azon kívüli foglalkozások **didaktikai feladatai** (új anyag feldolgozása, illetve alkalmazás, gyakorlás, ellenőrzés) szerinti csoportosításban szeretnénk ismertetni. A programok - szinte kivétel nélkül - használhatók az ismétléshez, összefoglaláshoz is, ezt külön nem jelezzük.

A közölt programok mindegyike közvetlenül kapcsolódik az általános iskolai tanterv törzsanyagához. A programok többségéből azonban ezen túlmutató összefüggések, következtetések is levonhatók. Ezeket kiegészítő anyagként, vagy fakultatív foglalkozáson, szakkörön javasoljuk hasznosítani. A tanterv és tanulócsoporthoz ismeretében, a körülmények mérlegelése alapján dönthető el, hogy a felhasználásra kerülő programból mennyit hasznosíthatunk a tanítási órán a törzsanyag feldolgozásához, s mit tudunk felhasználni kiegészítő anyagként, illetve fakultatív foglalkozáson és szakkörön. Mindezt figyelembe véve, a módszertani javaslatokhoz kapcsolva nem jelöljük a fenti felhasználási területeket.

A módszertani javaslatokat az egyes didaktikai feladatokon belül a programok funkciójára (kísérlet, mérés, szimulálás stb.) és szerkezetére szerinti csoportosításban közöljük. A hasonló funkciójú és szerkezetű programok közül csak egy-egy program használatához adunk viszonylag részletesebb módszertani javaslatot, alcímként kiemelve a program címét, ezt követően jelezzük a többi, hasonló jellegű program alkalmazásának speciális lehetőségeit. Az egyes programok felhasználásához adott javaslatokon belül - x - jelzéssel választjuk el a viszonylag önálló gondolati egységeket, alkalmazási területeket.

A módszertani javaslatok megfogalmazásában nem törekedhetünk a teljességre. Bízunk abban, hogy az ajánlott módok újabb lehetőségek kimunkálására adnak ösztönzést a programok felhasználóinak körében is.

### 1.3.1. Javaslatok az új ismeretek feldolgozásához

A számítógépet az új ismeretek feldolgozásának fázisában kísérletek bemutatására, a mérések pontosabbá tételére, a kísérletek szimulálására és az összefüggések felismertetésére használhatjuk fel.

#### 1.3.1.1. Kísérletek bemutatása számítógéppel

A Commodore 16-os (Plus/4-es) számítógépet az általános iskolai fizikatanításban a rezgések és színek tanításakor alkalmazhatjuk kísérleti eszközként.

##### a/ HANGGENERÁTOR

A 8. osztályban megismerkednek a tanulók a rezgőmozgás alapvető jellegzetességeivel, a rezgésszám fogalmával, mértékegységével (Hz). A hullámmozgásra vonatkozó ismeretek feldolgozása keretében tanulnak a hangról, a hang terjedéséről, s arról, hogy a két közel egyenlő rezgésszámú hangvilla által keltett hullám hol erősíti, hol gyengíti egymást, ezért halljuk lüktetve a hangvillák hangját (hanglebegés).

Ezzel kapcsolatban a HANGGENERÁTOR című program alkalmazásakor szemléltetni tudjuk a hangforrás rezgésszáma és a hallott hang magassága közötti kapcsolatot, bemutathatjuk a hanglebegést és a különböző hangközöket.

A program betöltése után megszólaltatjuk a 440,3 Hz rezgésszámú hangot, ami jó megközelítéssel a normál "a" hangnak (440 Hz) felel meg. (Programozási okok miatt nem lehet pontosan előállítani 440

Hz-es rezgésszámot.) Amikor a hangverseny előtt "hangol" a zenekar, akkor az első hegedűs mindig ezt a normál "a"-t adja meg társai számára, ehhez hangol a többi zenekari tag. (Csak 1939. óta 440 Hz a normál "a" rezgésszáma a nemzetközi megállapodás szerint, korábban 435 Hz volt.)

Növeljük a rezgésszámot. Miközben egyre magasabb hangot hallunk, a képernyőről leolvashatjuk a hallott hang rezgésszámát. Minél nagyobb a hangforrás rezgésszáma, annál magasabb hangot hallunk. Érdekességként megemlíthetjük, hogy a rádió minden reggel 1000 Hz rezgésszámú "beállítóhangot" sugároz a sztereoadásokhoz. Ezt is megszólaltathatjuk.

- x -

A hanglebegést nagyon változatos formában és meggyőző módon tudjuk szemléltetni a számítógéppel. Mindkét hangforrást bekapcsoljuk a kiinduló rezgésszámmal (440,3 Hz). Célszerű megfigyeltetnünk a tanulókkal, hogy amikor a második hangforrást bekapcsoljuk, úgy tűnik, mintha csak a hangerőt változtattuk volna meg. Annyira "egybehangzó" a két hang, hogy nem is lehet megkülönböztetni azokat egymástól.

Ha változtatjuk az egyik hangforrás rezgésszámát, azon nyomban megkülönböztethető a két hang. Azt is megfigyeltethetjük, hogy most nem egyenletes erősségű a hang, hanem egyenlő időközökben, váltakozva erősödik és gyengül (hanglebegés). Ha fokozatosan növeljük a két hangforrás rezgésszáma közötti különbséget, akkor változik a hallott hang erősödése, gyengülése közötti időköz, mivel a két rezgésszámtól függően, más-más időközben erősítik, gyengítik egymást a hanghullámok.

- x -

Azt is bemutatathatjuk, hogy milyen a két hangforrás rezgésszáma közötti arány, ha oktáv, terc vagy kvint hangköznek megfelelően szólaltatunk meg két hangot. (Koordináció az ének-zenével.) Az egyik hangforrást alaphelyzetben hagyjuk, ennek megfelelően az 440,3 Hz rezgésszámú hangot ad. A másik hangforrást 880,6 Hz-re hangoljuk. (Csak a 880-at kell "beütni".) Jól, tisztán halljuk a megközelítően az  $a^1$ -nek és az  $a^2$ -nek megfelelő, oktáv távolságú, két hangot. A rezgésszámok aránya az  $a^2$  és az  $a^1$  között 2 : 1. Hasonló módon érzékeltethetjük a terc (harmad) hangtávolságot. Ebben az esetben a rezgésszámok aránya az ún. "tiszta hangolásban" 5 : 4. A számítógépen az 550,9 Hz-et állíthatjuk be a 440,3 Hz mellé. A kvint (ötöd) hangtávolság esetén 3:2 a rezgésszámok aránya. A számítógépen a 661,8 Hz-et állíthatjuk be a 440,3 Hz mellé.

## h/SZINEK

A tanulók a 8. osztályban tanulnak arról, hogy a testek színe attól függ, hogy a testről milyen fény jut a szemünkbe. Egyszerű kísérletek segítségével érzékeltethetjük a fehér fény színekre történő felbontását, valamint a színek "összetételét".

A számítógéppel a SZINEK című program felhasználásával szemléletesen bemutathatjuk a különféle színek összetételét, ha színes monitor (TV-készülék) áll rendelkezésünkre.

Alaphelyzetben három különböző színű téglalapot látunk a képernyőn. Ha úgy közelítjük egymáshoz a három téglalapot, hogy azok fokozatosan egyre jobban fedjék egymást, akkor a két-két szín "keverékeként" létrejövő színeket látjuk. A három szín összetételeként pedig fehér színt látunk.

### 1.3.1.2. A mérési eredmények pontosabbá tétele

A C-16-os (Plus/4-es) számítógéppel lehetőség adódik arra, hogy - megfelelő kiegészítő eszközök alkalmazásával - az eddigi, általánosan elterjedt módszerhez viszonyítva, pontosabb mérési eredményeket kapjunk a különböző, mozgásokkal kapcsolatos kísérletek során. Könyvünk a 8. osztályban tárgyalt szabadesés vizsgálatához tartalmaz programot.

#### SZABADESÉS MÉRÉSE

A mérés elvégzéséhez olyan kísérleti eszköz szükséges, amelynek segítségével egy vasgolyót ejthetünk le különböző magasságokból, s közben mérhetjük az időt a számítógép segítségével az elengedés pillanatától a földreérés pillanatáig. (Az eszközzel kapcsolatos további részletek a program előtti ismertetőben olvashatók.)

Kiindulásul a kísérleti összeállítás vázolata jelenik meg a képernyőn. A gép segítségével - e vázlat felhasználásával - szimulálhatjuk is a kísérletet.

Lényeges, hogy a tanulók e vázlat és a hozzáfűzött magyarázatunk alapján világosan lássák: a kísérlet során a golyó elengedésétől a földreérés pillanatáig eltelt időt mérjük a számítógéppel századmásodpercnyi pontossággal.

A kísérlet elvégzésének, elemzésének egyik módja, hogy 10 cm-kénti különbséggel, egyre magasabbról ejtjük le az acélgolyót, és a megtett utat (s), a mért időt (t) táblázatba foglaljuk, majd kiszámítjuk a 10 - 10 cm-es út megtételéhez szükséges időt (t).

s	t	$\Delta t$
0	0	0,14 s
0,1 m	0,14 s	0,06 s
0,2 m	0,20 s	0,05 s
0,3 m	0,25 s	0,04 s
0,4 m	0,29 s	0,03 s
0,5 m	0,32 s	

A táblázatból kitűnik, hogy az azonos (0,1 m) hosszúságú utak megtételéhez szükséges idő egyre kisebb (0,14 s, 0,06 s, stb.) Ebből következik, hogy a szabadon eső golyó sebessége egyre nagyobb, vagyis gyorsuló mozgást végez.

- x -

A kísérlet elvégzésének és elemzésének másik módja, hogy a 10 cm-kénti méréseket 90 cm-ig folytatjuk, s az így felvett táblázat alapján keresünk összefüggést az út és az idő között:

<u>s</u>	<u>t</u>
0	0
0,1 m	0,14 s
0,2 m	0,20 s
0,3 m	0,25 s
0,4 m	0,29 s
0,5 m	0,32 s
0,6 m	0,35 s
0,7 m	0,38 s
0,8 m	0,41 s
0,9 m	0,43 s

A táblázat szerint 0,14 s alatt tette meg az acélgolyó az első 10 cm-nyi utat. Keressük ki a táblázatból, mekkora utat tett meg a golyó 2-szer, 3-szor annyi idő alatt! Azt találjuk, hogy a golyó 0,29 s alatt 0,4 m utat, 0,43 s alatt pedig 0,9 m utat tett meg. Vagyis a golyó 2-szer, 3-szor annyi idő alatt 4-szer, 9-szer akkora utat tett meg.

Célszerű grafikusan is ábrázolni a megtett utakat és hozzájuk tartozó időtartamokat. A grafikonról is jól megállapítható, hogy a szabadon eső test egyenlő időközök alatt egyre nagyobb utakat tesz meg.

### 1.3.1.3. A jelenségek szimulálása

A közölt programok egy része segítségével mozgó sematikus ábrákon megjeleníthetjük a fizikai jelenségeket. E szimulációs programok többsége grafikonon is rögzíti a jelenségre jellemző mennyiségek közötti összefüggést. Elsődlegesen a jelenség kísérleti bemutatását követően javasoljuk e programok alkalmazását. Bemutatásuk azonban semmiképpen sem pótolhatja a kísérletek tényleges elvégzését. Ezek a programok igen hatékonyan segíthetik az ismeretek megértését, rögzítését.

#### a/ ÜTKÖZÉSEK

A lendület megmaradása a 8. osztályban tananyag. A rendelkezésre álló eszközökkel vagy csak kvalitatív kísérletet tudunk bemutatni, vagy viszonylag nagyszámú logikai lépés után tudunk eljutni a törvény megfogalmazásáig. Az ÜTKÖZÉSEK című program segítséget nyújt a kísérleti tapasztalatok elemzéséhez, a felismertett összefüggés elsődleges megerősítéséhez.

A program azt a kísérletet szimulálja, amelyben egy mozgó test egy másik, álló testnek ütközik. Mi választhatjuk meg, hogy mekkora legyen a két test tömege és a mozgó test sebessége.

Célszerű a bemutatást először egyenlő (pl. 12-12 kg) tömegű testekkel végezni, mivel ez a tanulók által legjobban ismert eset, s egyszerű a számítás. Javasoljuk az első néhány alkalommal viszonylag kis sebességet (pl. 2 m/s-et) megadni, mivel így több idő jut a sematikus ábra értelmezésére. Ajánlatos a bemutatást nagyobb sebességgel (pl. 10 m/s-sel) is megismételni.

Az eddig vizsgált esetekben azt láttuk, hogy a mozgó test az ütközés után megáll, az álló test megindul, és sebessége ugyanakkora lesz az ütközés után, mint a másiké volt az ütközés előtt.

Célszerű a gép által kiírt adatokat feljegyeztetni a tanulókkal, majd kiszámíttatni, hogy mennyi a két test együttes lendülete az ütközés előtt és az ütközés után. A két mennyiség egyenlő. Az eddigi példákban:

$$24 \text{ kg} \cdot \text{m/s} = 24 \text{ kg} \cdot \text{m/s},$$

$$120 \text{ kg} \cdot \text{m/s} = 120 \text{ kg} \cdot \text{m/s}.$$

- x -

Következő fokozatként javasoljuk a mozgó test tömegét az eredeti 2-szeresére, 3-szorosára növelni, az álló test sebességét változatlanul hagyva. A sebességet 3 m/s-ra változtatjuk. Megfigyeltetjük a tanulókkal, hogy most ütközés után nem áll meg a mozgó test, hanem tovább megy, az eredetinel kisebb sebességgel. Érdekes, hogy a kisebb tömegű test sebessége nagyobb lesz a másik test eredeti sebességénél:

$$m_1 = 24 \text{ kg} \quad m_2 = 12 \text{ kg}$$

$$m_1 = 36 \text{ kg} \quad m_2 = 12 \text{ kg}$$

$$v_1 = 3 \text{ m/s} \quad v_2 = 0 \text{ m/s}$$

$$v_1 = 3 \text{ m/s} \quad v_2 = 0 \text{ m/s}$$

$$u_1 = 1 \text{ m/s} \quad u_2 = 4 \text{ m/s}$$

$$u_1 = 1,5 \text{ m/s} \quad u_2 = 4,5 \text{ m/s}$$

Ha kiszámíttatjuk a tanulókkal a két test együttes lendületét az ütközés előtt és az ütközés után, akkor a következőket kapjuk:

$$72 \text{ kg} \cdot \text{m/s} = 72 \text{ kg} \cdot \text{m/s}$$

$$108 \text{ kg} \cdot \text{m/s} = 108 \text{ kg} \cdot \text{m/s}$$

- x -

Megfigyeltethetjük a tanulókkal, hogy minél nagyobb a mozgó test tömege az állóhoz viszonyítva, annál nagyobb arányban nő az álló test sebessége az ütközés után a mozgó test eredeti sebességéhez képest. (Az előző példában 4 m/s : 3 m/s, illetve 4,5 m/s : 3 m/s.) Kérdés: Mennyire növelhető ez az arány?

Ha a program által megszabott legnagyobb tömegűnek (10000 kg-nak) választjuk a mozgó testet és 1 kg tömegűnek az álló testet, akkor a következőket kapjuk:

$$m_1 = 10000 \text{ kg} \quad m_2 = 1 \text{ kg}$$

$$v_1 = 3 \text{ m/s} \quad v_2 = 0 \text{ m/s}$$

$$u_1 = 2,99 \text{ m/s} \quad u_2 = 5,99 \text{ m/s}$$

A két test sebességének aránya ( $5,99 \text{ m/s} : 3 \text{ m/s}$ ) megközelíti a 2-t. Ha végtelen nagyra választanánk a mozgó test tömegét, az álló test sebessége akkor lenne 2-szerese a másik test sebességének. A kisebb tömegű, álló test sebessége tehát nem növelhető tetszés szerinti mértékben, ahogy ezt gondolhatnánk az első két példa alapján.

Érdekes azt is észrevetetnünk a tanulókkal, hogy miként változik az ütköző testek sebessége, ha növeljük a mozgó test sebességét, minden más változatlanul hagyása mellett:

$m_1 = 36 \text{ kg}$	$m_2 = 12 \text{ kg}$	$m_1 = 36 \text{ kg}$	$m_2 = 12 \text{ kg}$
$v_1 = 3 \text{ m/s}$	$v_2 = 0 \text{ m/s}$	$v_1 = 6 \text{ m/s}$	$v_2 = 0 \text{ m/s}$
$u_1 = 1,5 \text{ m/s}$	$u_2 = 4,5 \text{ m/s}$	$u_1 = 3 \text{ m/s}$	$u_2 = 9 \text{ m/s}$

Az adatok tanúsága szerint, a második esetben az eredeti kétszeresére nőtt a testek sebessége az ütközés után. A két test együttes lendülete az ütközés előtt és után:

$$108 \text{ kg}\cdot\text{m/s} = 108 \text{ kg}\cdot\text{m/s}, \qquad 216 \text{ kg}\cdot\text{m/s} = 216 \text{ kg}\cdot\text{m/s}.$$

- x -

Felvethetjük azt a problémát is, hogy mi történik akkor, ha a mozgó test tömege kisebb, mint az álló test tömege. (A program futtatása előtt célszerű véleményt kérni a tanulóktól.)

Két példa:

$m_1 = 12 \text{ kg}$	$m_2 = 24 \text{ kg}$	$m_1 = 12 \text{ kg}$	$m_2 = 36 \text{ kg}$
$v_1 = 3 \text{ m/s}$	$v_2 = 0 \text{ m/s}$	$v_1 = 3 \text{ m/s}$	$v_2 = 0 \text{ m/s}$
$u_1 = -1 \text{ m/s}$	$u_2 = 2 \text{ m/s}$	$u_1 = -1,5 \text{ m/s}$	$u_2 = 1,5 \text{ m/s}$

Ütközés után az álló test megindul, a kisebb tömegű mozgó test pedig ellenkező irányba halad, mint eredetileg. Ezt az "ellenkező" irányt jelzi a negatív előjelű sebességadat. Érdekességként megfigyelhetjük, hogy amikor az álló test tömege 3-szor akkora, mint a mozgó test tömege, akkor ütközés után egyenlő a két test sebességének abszolút értéke ( $1,5 \text{ m/s}$ ).

Ha most kiszámíttatjuk a tanulókkal a két test ütközés előtti és ütközés utáni együttes lendületét, akkor természetesen az előjelek figyelembe vételével kell számolniuk:

$$36 \text{ kg}\cdot\text{m/s} = 36 \text{ kg}\cdot\text{m/s}, \qquad 36 \text{ kg}\cdot\text{m/s} = 36 \text{ kg}\cdot\text{m/s}.$$

- x -

#### b/ FÉNYTERJEDÉS

A tanulók a 8. osztályban ismerkednek meg azzal, hogy az új közeg határához érkező fénysugár egy része visszatér a régi közegbe, más része pedig behatol az új közegbe. A fényvisszaverődés és a fénytörés jelenségét általában az optikai paddal mutatjuk be a



levegő és az üveg (műanyag) esetére, összehasonlítva a beesési szöget a visszaverődési szöggel, illetve a törési szöggel. Körülményesebbé válik azonban a kísérlet akkor, ha a levegő és a víz (vagy más anyagok) alkalmazásával akarjuk a jelenséget bemutatni, az anyagi minőségtől való függést érzékelteni.

A FÉNYTERJEDÉS című program segítségével szemléletesen lehet szimulálni a fényvisszaverődés és a fénytörés jelenségét a levegő és a víz, a levegő és az üveg, a levegő és a gyémánt, illetve a víz és a gyémánt esetében. A program a beesési szög megadása után kirajzolja a fény útját, s kiírja a beesési szög, a visszaverődési szög és a törési szög nagyságát.

Először annak bemutatását javasoljuk, hogy miként változik a visszaverődési szög és a törési szög, ha fokozatosan növeljük a beesési szöget. A képernyőn megjelenő rajz és a kiírt adatok alapján megfigyelhetjük, hogy amint nő a beesési szög, nő a visszaverődési szög is.

Következő lépésként észrevetetjük, hogy

- a visszaverődési szög mindegyik esetben egyenlő a beesési szöggel,
- a törési szög mindegyik esetben kisebb, mint a beesési szög (megjegyezve, hogy a vizsgált esetekben optikailag ritkább közegből lép a fény optikailag sűrűbb közegbe).

Olyan megoldást is választhatunk a fentiek felismertetéséhez, hogy táblázatba foglaljuk a képernyőn megjelenő adatokat, s ez alapján végeztetjük az összehasonlítást.

Például a levegő és a víz esetében a következőket kapjuk:

Beesési szög	10°	20°	30°	40°	50°	60°	70°	80°
Visszaverődési szög	10°	20°	30°	40°	50°	60°	70°	80°
Törési szög	7°	14°	22°	28°	35°	40°	44°	47°

A táblázat alapján azt is érzékeltehetjük, hogy a beesési szöggel egyidejűleg nő ugyan a törési szög, de nem egyenes arányban. Ha például 6-szorosára, 8-szorosára nő a beesési szög a 10°-hoz képest, nem ugyanilyen arányban nő a törési szög.

- x -

A program jó lehetőséget nyújt annak felismertetéséhez is, hogy a törési szög függ a két közeg anyagi minőségétől. Bár általános iskolában nem tanítunk a törésmutatóról, a képernyőn megjelenő adatok táblázatba foglalása, elemzése révén bizonyos fokú tapasztalati előkészítést adhatunk ehhez.

Ha táblázatba foglaljuk a különböző anyagpárok esetén a beesési és a törési szögek nagyságát, érdekes összehasonlítást tehetünk:

Beesési szög		15°	30°	45°	60°	75°
Töré- si szög	levegő - víz	11°	22°	32°	40°	46°
	levegő - üveg	9°	19°	28°	35°	40°
	levegő - gyémánt	6°	11°	16°	20°	23°
	víz - gyémánt	8°	15°	22°	28°	32°

Ha azonos beesési szöggel érkezik a fény az új közeg határához, más-más a törési szög attól függően, hogy milyen anyagból milyen anyagba lép a fénysugár. A vizsgált négy eset közül azonos beesési szög esetén legnagyobb a törési szög, ha a fény a levegőből a vízbe lép, a legkisebb akkor, ha a levegőből a gyémántba lép. Az első három esetben a levegőből lép a fénysugár valamilyen más anyagba. Attól függően, hogy mi ez a második anyag, más-más a törés szöge.

- x -

### c/ LENCSEK

A lencsék képalkotásaival a 8. osztályban ismerkednek meg a tanulók, részben a demonstrációs bemutatót szolgáló optikai paddal, részben a fénytani tanuló kísérleti készlettel végzett kísérletek révén. A LENCSEK című program e kísérletek szimulálását teszi lehetővé, biztosítva a képalkotások közötti hasonlóságok és eltérések egyszerű felismerését, valamint a képszerkesztésekhez alkalmazott "nevezetes sugármenetek" gyors, pontos szemléltetését. Először azt célszerű bemutatni, hogy miként változik a domború (gyűjtő-) lencsében keletkező kép (helye, nagysága, minősége), ha a tárgyat közelítjük a lencséhez. E bemutatáshoz 10-12 cm-es fókusztávolságot célszerű választani, mivel ebben az esetben lehet folyamatosan vizsgálni a képet (anélkül, hogy az a képernyőn kívül kerülne).

A tanulók jól megfigyelhetik, hogy amint közeledik a tárgy a lencséhez, a kép távolodik a másik oldalon, és egyre nagyobb lesz. Amikor a tárgy a fókusztávolság kétszeresében van, akkor a kép a másik oldalon ugyancsak a fókusztávolság kétszeresében van, s nagysága egyenlő a tárgy nagyságával. Érdekes közben a gép által kiírt adatokat is összehasonlítani.

Ha tovább közelítjük a tárgyat a lencséhez, akkor a kép a másik oldalon már a fókusztávolság kétszeresén kívül keletkezik, s nagyobb lesz, mint a tárgy. Célszerű lassú lépésekben mutatni azt a helyzetet, amikor a tárgy egészen közel kerül a fókuszhhoz. A kép ebben az esetben nagymértékben nő, és távolodik a lencsétől. Mikor a tárgy a fókuszpontba kerül, "eltűnik" a kép, és a képtávolság adata helyett megjelenik a felírás, hogy "nincs kép".

A tárgy további közelítésekor újra megjelenik a nagyított kép, de most már - a korábbiaktól eltérően - a tárggyal azonos oldalon, látszólagos és a tárggyal megegyező állású. A látszólagos kép helyzetéből adódóan a képtávolságot most negatív előjelű mennyiség jelzi.

Jól szemléltethető a program segítségével, hogy azonos fókusz-távolság és tárgy-távolság esetén a nagyobb tárgyról nagyobb kép keletkezik.

A képszerkesztéshez használatos "nevezetes sugármenetek" szemléltetéséhez célszerű nagyobb fókusz-távolságot (pl. 40 cm-t) választanunk. Ha a tárgyat a lehető legnagyobbra állítjuk, és a tárgy-távolságot elég nagyra (pl. 150 cm-re) választjuk, akkor jól áttekinthető képet kapunk, ugyanakkor nem marad le semmi a képernyőről.

Célszerű a "nevezetes sugármeneteket" először egyenként bemutatni, majd utána mindegyiket együtt. Hívjuk fel a tanulók figyelmét arra, hogy azért célszerű a sok-sok fénysugár közül éppen ezeket kiválasztani a kép megszerkesztéséhez, mert könnyű, egyszerű ezek megrajzolása, a beesési és törési szög ismerete nélkül. (Ha "tetszőlegesen" választanánk meg a szerkesztéshez a fénysugarakat, akkor a beesési szög figyelembe vételével meg kellene határoznunk a törési szöget, majd ennek megfelelően kellene a fény további útját megrajzolnunk.)

- x -

A TÜKRÖK című program felhasználásával a homorú és a domború tükrök képalkotásait elemezhetjük a lencsék esetében alkalmazott módszerekhez hasonlóan.

#### d/ EGYENLETES MOZGÁS

A testek mozgásának vizsgálata a 8. osztályban tananyag. A jelenségek bemutatását a Mikola-csővel, a demonstrációs készlethez tartozó kénporos lejtővel és kiskocsikkal végezzük. A tapasztalatok további elemzését teszik lehetővé azok a programok, amelyek szimulálják a különféle mozgásfajtákat, s ezzel egyidejűleg grafikonon ábrázolják az út és az idő közötti összefüggéseket.

Nagy előnye a grafikonok képernyős megjelenítésének, hogy konkrét mennyiségpárok ismeretében, a tanulók szemeláttára alakul ki a grafikon, egyszerű ezt a folyamatot lassítani, gyorsítani, egy-egy lépésnél leállítani, gyorsan lehet új adatpárokkal másik grafikont rajzoltatni, az előzővel összehasonlítani.

Az EGYENLETES MOZGÁS című program véletlenszerűen választja meg a kis négyzettel ábrázolt test sebességét. Indítás után függőleges irányban mozog a test. A program lehetővé teszi, hogy nyomképpel vagy anélkül futtassuk a programot. A nyomkép kirajzoltatásának az az előnye, hogy a jelenség lejátszódása után jól látható, mekkora utakat tett meg a test azonos idők alatt. A nyomkép nélküli bemutatásnak pedig az az előnye, hogy az közelebb áll a kísérletben ténylegesen látott jelenséghez.

Azt javasoljuk, hogy először nyomkép kirajzoltatása nélkül végezzük a bemutatást. Az elemzéshez viszont célszerű a nyomképeket is kirajzoltatnunk. Ha nagyobb a test sebessége, akkor távolabb vannak a kirajzolódó nyomképek egymástól, mint kisebb sebesség esetén.

A test mozgásával egyidejűleg az előre felrajzolt koordinátarendszerbe az út és az idő összefüggésének megfelelően pontokat rajzol a gép. E pontok összekötéseként lépésenként vagy folyamatosan folytonos vonallal kirajzoltatható az adott mozgásnak megfelelő grafikon.

A grafikon lépésenkénti kirajzoltatása különösen jó lehetőséget nyújt a megadott sebességadat értelmezésére. A 10 m/s sebesség esetében például leolvasható a kialakuló grafikonról, hogy a 20 m-es úthoz 2 s, a 30 m-es úthoz 3 s, a 40 m-es úthoz 4 s idő tartozik. E két-két mennyiség hányadosa mindegyik esetben 10 m/s. Két vagy több grafikon megrajzoltatása után azt is megfigyeltethetjük a tanulókkal, hogy a tanulmányozott egyenesvonalú egyenletes mozgást ábrázoló grafikon mindegyik esetben az origóból kiinduló félegyenes. Minél nagyobb a test sebessége, annál "meredekebb" e mozgást ábrázoló félegyenes.

- x -

#### e/ MOZGÁS A LEJTŐN

Az egyenesvonalú egyenletesen gyorsuló mozgást az általános iskolában lejtőn leguruló golyóval vagy kiskocsival szoktuk szemléltetni. A szerzett tapasztalatok elemzését, elmélyítését segíti a MOZGÁS A LEJTŐN című program.

A program 1-45°-os lejtő rajzoltatását teszi lehetővé. Ezen mozog a testet ábrázoló kis négyzet. A programot első ízben most is a nyomképek kirajzolása nélkül, majd ezt követően a nyomképek kirajzolásával javasoljuk futtatni. A kirajzolódó négyzetek növekvő távolságából jól lehet következtetni a test gyorsuló mozgására.

Tanulságos számbavennünk azt is, hogy mennyi a különböző hajlásszögű lejtőn kirajzolódó nyomképek száma, mivel a gép a nyomképeket egyenlő időközönként rajzolja ki az azonos hosszúságú lejtőre. Így a nyomképek száma arányos az eltelt idővel.

Például:

A lejtő hajlásszöge	10°	20°	30°	40°
A nyomképek száma	16	12	10	8

Minél nagyobb a lejtő hajlásszöge, annál kisebb az azonos időközönként kirajzolódó nyomképek száma, vagyis annál rövidebb idő szükséges az azonos út megtételéhez.

A lejtőn mozgó test szimulációjával egyidejűleg a képernyőn az előre felrajzolt koordinátarendszerbe felrajzolja a gép az út-idő grafikonot. Jó elemzési lehetőséget nyújt a különböző hajlásszögű lejtőn történő gyorsuló mozgást ábrázoló grafikonok összehasonlítása.

#### f/ GYORSULÓ MOZGÁS

A tanulók a 8. osztályban a test lejtőn való mozgásának a tanulmányozása alapján jutnak el a gyorsuló mozgás fogalmához, s ezen belül az egyenletesen gyorsuló mozgás néhány jellegzetességének a megismeréséhez. Tanulják, hogy ha egy test sebessége időegységről időegységre ugyanakkora értékkel változik, akkor mozgása egyenletesen változó mozgás. A gyorsulást első megközelítésben olyan mennyiségként értelmezik, hogy az megmutatja, mekkora az egyenletesen változó mozgást végző test egy

másodperc alatt bekövetkező sebességváltozása. Az egyenletesen változó mozgásra jellemző, hogy gyorsulása állandó. A GYORSULÓ MOZGÁS című program segíti a bevezetett új fogalmak, mennyiségek megértését, a köztük levő kapcsolatok felismerését.

Először most is nyomjelek nélkül, majd nyomjelek kirajzoltatásával javasoljuk a gyorsuló mozgás érzékeltetését a program segítségével. Jól megfigyelhető a képernyőn, hogy egymástól egyre nagyobb távolságra rajzolódnak ki a testet ábrázoló kis négyzetek, jelezve, hogy a test egyre nagyobb utakat tesz meg egyenlő időközök alatt.

A program a G billentyű megnyomásakor a grafikon kirajzolásával egyidejűleg kiírja a test által megtett utat, az időt és a pillanatnyi sebességet.

A tapasztalatok azt mutatják, hogy gondot okoz a tanulók jelentős részének annak megértése, hogy mit is jelent az egyenletesen változó mozgás kifejezésben az **egyenletes** szó, éppen a **változó** mozgás jelzőjeként. A program jó lehetőséget ad e kifejezések értelmezéséhez. A képernyőn egymás után megjelenő adatok felhasználásával táblázatba foglaljuk, hogy mekkora sebességre tesz szert a test az 1., a 2., a 3.,... a 6. másodperc végéig. Az egymást követő sebességadatokból kiszámítjuk, hogy mennyi az egy-egy másodperc alatt bekövetkező sebességváltozás.

Két táblázat a program alapján nyert adatok felhasználásával:

Idő	Sebesség	Sebesség- különbség 1 s alatt	Idő	Sebesség	Sebesség- különbség 1 s alatt
0	0		0	0	
		4 m/s			3,2 m/s
1. s	4 m/s	4 m/s	1. s	3,2 m/s	3,2 m/s
2. s	8 m/s	4 m/s	2. s	6,4 m/s	3,2 m/s
3. s	12 m/s	4 m/s	3. s	9,6 m/s	3,2 m/s
4. s	16 m/s	4 m/s	4. s	12,8 m/s	3,2 m/s
5. s	20 m/s	4 m/s	5. s	16,0 m/s	3,2 m/s
6. s	24 m/s		6. s	19,2 m/s	

Érdekes kiszámíttatnunk a tanulókkal a fenti adatpárokból a test által 1, 2, 3,... 6 másodperc alatti **átlagsebességet**, és ezt összehasonlíttatnunk az előző táblázatokban szereplő **pillanatnyi sebességekkel**. (Az átlagsebesség mindegyik esetben fele a pillanatnyi sebességnek.)

A képernyőn kirajzolódó görbékről megállapítható, hogy az a görbe "meredekebb", amelyik a nagyobb gyorsulással mozgó test mozgását jellemzi.

## g/ ÁTLAGSEBESSÉG

Az átlagsebességet a tanulók a 8. osztályban úgy értelmezik, mint azt a sebességet, amellyel a test egyenletesen mozogva ugyanazt az utat ugyanannyi idő alatt tenné meg, mint változó mozgással. A fogalom megértéséhez, értelmezéséhez nyújt segítséget az ÁTLAGSEBESSÉG című program

Indítás után váltakozva gyorsuló és lassuló mozgással, függőleges irányban halad a testet jelképező kis négyzet. Ezzel egyidejűleg a koordinátarendszerbe az út és az idő pillanatnyi értékeinek megfelelő pontokat rajzol a gép. E pontokat lépésenként vonallal köttethetjük össze. Minden egyes lépés esetén kiírja a gép az indulástól számított átlagsebességet.

Tanulságos megfigyeltetnünk a tanulókkal, hogy miként változik ez az átlagsebesség, ha azt a test mozgásának egyre nagyobb szakaszaira határozzuk meg. Különösen azoknak a mozgásoknak a tanulmányozása érdekes, amelyekben felváltva szerepel sebességnövekedés és -csökkenés. Ebben az esetben először növekszik a kiírt átlagsebesség, majd csökken.

Több grafikon elemzése után azt a feladatot is adhatjuk a tanulóknak, hogy a grafikon egy-egy soronkövetkező szakaszának kirajzolása előtt becsüljék meg a kirajzolt pontok alapján, hogy miként változik az átlagsebesség.

## h/ REZGŐMOZGÁS

A 8. osztályos tananyagban szerepelnek a rezgőmozgással kapcsolatos alapismeretek. A tanulmányozáshoz szükséges kísérletek elvégezhetők a mechanikai tanuló-kísérleti készlet eszközeivel és a demonstrációs készlettel. A rezgési idő, a rezgésszám és a rezgés tágassága fogalmának a kialakításához jól használható a REZGŐMOZGÁS című program.

A mozgást végző test most is kis négyzet képében jelenik meg, s 10 cm-re van az egyensúlyi helyzetből. Indítás után rezgőmozgást végez a test, s közben felrajzolja a program az út-idő grafikon néhány pontját. A program lefutása után megadja a gép a test távolságát az egyensúlyi helyzethez viszonyítva, s kiírja a rezgésidőt.

Tanulságos megfigyeltetnünk a tanulókkal a grafikon lassú, fokozatos kirajzoltatása közben, miként változik a test helyzete az egyensúlyi helyzethez mérten. Ezt nemcsak a kirajzolódó vonal alapján követhetjük nyomon, hanem a gép által kiírt, fokozatosan változó adatokból is.

A rezgésidő értelmezését oly módon végeztethetjük, hogy megfigyeltetjük a tanulókkal, hol éri el a gép által felrajzolt görbe ismét a maximumot, majd megállapítatjuk az e ponthoz tartozó időt. (A rezgés a maximumból indul.) Helyes grafikonleolvasás esetén a megállapított érték egyezik a gép által kiírt rezgésidővel. Célszerű a rezgésidőt más, azonos rezgésállapotú helyzetek közötti időkülönbség alapján is megállapítatni.

Javasoljuk más rezgésidőjű görbét is kirajzoltatni a géppel, meghagyva az előző görbét, hogy összehasonlíthassuk a különböző rezgésidőjű, de azonos tágasságú (amplitudójú) rezgések jellemzőit.

A program indítása előtt állítható a rezgés tágassága, maximálisan 40 cm-re. Tanulságos a különböző tágasságú rezgések megfigyeltetése és ezek grafikonjainak elemzése is. A grafikonok elemzéséhez kapcsolódóan javasoljuk a következő (vagy ehhez hasonló) kérdések feltevését:

- Milyen módon lehet elérni a valóságban, hogy növekedjék a test rezgési ideje?
- Hogyan lehet növelni a test rezgésének a tágasságát?

#### 1/ MELEGEDES-HŰLÉS

A 6. osztályban tananyag a termikus kölcsönhatás. Az I. témakörön belül először azt ismertetjük fel a tanulókkal, hogy a két, egymással érintkező, különböző hőmérsékletű test kölcsönösen hat egymásra. Ez az állapotváltozás addig tart, amíg a két test hőmérséklete egyenlő nem lesz. A II. témakör anyagának feldolgozása során, a belső energia és a fajhő fogalmának a megismerése lehetővé teszi annak felismerését is, hogy termikus kölcsönhatás közben ugyanannyival nő az egyik test energiája, mint amennyivel csökken a másik test energiája.

A termikus kölcsönhatás I. témakörön belüli elemzéséhez használhatjuk a MELEGEDES-HŰLÉS 1. című programot. Az ismeretszerzés adott szintjének megfelelően, a program két azonos anyagú (azonos fajhőjű) test hőmérséklet-változásának a vizsgálatát teszi lehetővé. Mivel a program futtatása sokkal rövidebb időt vesz igénybe, mint a kísérlet elvégzése, lehetőség nyílik arra, hogy a kísérleti tapasztalatok mellett több, különböző feltételeknek megfelelő hőmérséklet-változást elemezzünk.

Először célszerű egyenlőnek választani a két test tömegét, s úgy megadni a két kiinduló hőmérsékletet, hogy egyszerű legyen a számítás. (Mivel az alapul szolgáló kísérleteket általában vízzel végezzük, a hőmérsékletet  $0^{\circ}\text{C}$  és  $100^{\circ}\text{C}$  között lehet megadni.)

Indításkor a képernyőn kirajzolódik az egymással érintkező két test. Mindegyikben egy-egy hőmérő van. A hőmérőkön a higany szintje és egy-egy vízszintes vonalka mutatja a kiindulási hőmérsékletet. Alattuk a megadott hőmérséklet-adatok látszanak.

A képernyőn jól nyomon követhető, miként csökken az egyik test hőmérséklete, s miként nő a másik test hőmérséklete: ennek megfelelően változik a két hőmérőben a higanyszál hossza, és változik a kiírt két hőmérséklet-adat. Ezzel egyidejűleg kirajzolódik a két test hőmérséklet-változását ábrázoló grafikon.

Ha például 1-1 kg volt a két test tömege, s  $80^{\circ}\text{C}$  és  $20^{\circ}\text{C}$  a kiindulási hőmérséklet, akkor a kirajzolódó két görbe vonal szimmetrikus helyzetű, és  $50^{\circ}\text{C}$ -nál találkozik a két vonal. Ezt az adatot ki is írja a gép.

Vetessük észre a tanulókkal, hogy most egyenlő volt a két test tömege, s azonos volt az anyaguk, a kialakuló közös hőmérsékletet a két kiindulási hőmérséklet számtani közepeként lehet kiszámítani. Akkor is  $50^{\circ}\text{C}$  lesz a kialakuló közös hőmérséklet, ha  $100^{\circ}\text{C}$ -ot és  $0^{\circ}\text{C}$ -ot választunk kiindulásul.

(Megjegyzés. Ha nagyobb a két test tömege, akkor hosszabb idő szükséges a közös hőmérséklet kialakulásához. Ilyen esetben előfordulhat, hogy a grafikonon feltüntetett idő alatt nem "találkozik" a két test hőmérséklet-változását ábrázoló görbe vonal.

Ilyenkor a T billentyű megnyomása után folytatódik a grafikon rajzolása.)

Tanulságos a két test hőmérséklet-változását ábrázoló grafikonok kialakulását megfigyeltetni akkor, ha különböző a két test tömege. Ha például 2 kg-nak választjuk a  $100^{\circ}\text{C}$ -os test tömegét, s 0,5 kg-nak a  $0^{\circ}\text{C}$ -os test tömegét, akkor  $80^{\circ}\text{C}$  lesz a közös hőmérséklet. A kirajzolódó két görbe vonal most már nem szimmetrikus, a kialakuló közös hőmérséklet magasabb a két kiinduló hőmérséklet számtani közepénél.

Célszerű ezután azt a változatot is bemutatni, amikor az előzőhöz képest felcseréljük a megadott mennyiségeket: a 2 kg tömegű test hőmérséklete  $0^{\circ}\text{C}$ , a 0,5 kg tömegű test hőmérséklete pedig  $100^{\circ}\text{C}$ . Ebben az esetben  $20^{\circ}\text{C}$  lesz a kialakuló közös hőmérséklet, s ennek megfelelően alakul a kirajzolódó két görbe is.

E két esetben (s további más, hasonló esetekben is) azt tapasztaljuk, hogy a kialakuló közös hőmérséklet (azonos anyagok esetén) mindig a nagyobb tömegű test hőmérsékletéhez van közelebb. Emlékeztessük a tanulókat arra, hogy csak kis mértékben emelkedik a fürdőkádban levő hideg víz hőmérséklete, ha abba egy pohár meleg vizet öntünk. Vagy: csak kis mértékben csökken a mosogatóban levő meleg víz hőmérséklete, ha abba egy csészényi hideg vizet öntünk. Ezt követően javasoljuk a tömeg- és hőmérséklet-adatok megadása után a várható közös hőmérséklet megbecsültetését. Miután a tanulók feljegyezték a füzetükbe az általuk becsült közös hőmérsékletet, kirajzoltatjuk a géppel a grafikont, s összevetjük a kiírt hőmérsékletet a tanulók által becsült értékkel.

- x -

Megkönnyíthetjük a vizsgált hőmérséklet-változások áttekintését, a következtetések levonását, ha táblázatba foglaljuk a kiindulási adatokat és a kialakuló közös hőmérsékleteket.

Például:

Az egyik test		A másik test		A kialakuló közös hőmérséklet
tömege	hőmérséklete	tömege	hőmérséklete	
1 kg	$80^{\circ}\text{C}$	1 kg	$20^{\circ}\text{C}$	$50^{\circ}\text{C}$
1 kg	$100^{\circ}\text{C}$	1 kg	$0^{\circ}\text{C}$	$50^{\circ}\text{C}$
2 kg	$100^{\circ}\text{C}$	0,5kg	$0^{\circ}\text{C}$	$80^{\circ}\text{C}$
2 kg	$0^{\circ}\text{C}$	0,5kg	$100^{\circ}\text{C}$	$20^{\circ}\text{C}$
2 kg	$80^{\circ}\text{C}$	0,5kg	$20^{\circ}\text{C}$	$68^{\circ}\text{C}$
2 kg	$0^{\circ}\text{C}$	0,5kg	$80^{\circ}\text{C}$	$32^{\circ}\text{C}$

- x -

A 6. osztály I. témakörén belül a termikus kölcsönhatás tanulmányozásakor elsősorban azt vizsgáltuk, hogy miként változik a két test állapota (hőmérséklete). A II. témakör feldolgozásakor viszont a belső energia változásának a vizsgálata áll a középpontban. Ehhez nyújt segítséget a MELEGEDES-HÜLÉS 2. című program.

A belső energia fogalmának a kialakításához fontos összetevő annak vizsgálata, hogy miként változik meg valamely test belső energiája a különböző tényezőktől (hőmérséklet-változástól, tömegtől, fajhőtől) függően.



Ehhez szeretnénk a következőkben néhány módszertani javaslatot adni.

Először azt javasoljuk megvizsgálni a számítógép segítségével, hogy milyen mértékben emelkedik az 1 kg tömegű 0°C-os víz hőmérséklete, ha az egy másik edényben levő 20°C, 40°C, 60°C, 80°C, majd 100°C hőmérsékletű vízzel kerül kölcsönhatásba. Más szavakkal: azt vizsgáljuk, hogy milyen "melegítő hatása" van a különböző hőmérsékletű víznek.

A megfelelő adatok beírása és a program indítása után jól látható az egymást követően kirajzolódó grafikonokon, hogy miként lesz egyre magasabb a kialakuló közös hőmérséklet, miként melegedik fel egyre jobban a változatlan tömegű, azonos kiindulási hőmérsékletű víz. Ha a megadott és a képernyőről leolvasott adatokat táblázatba foglaljuk, akkor a következőket kapjuk:

Allandó a két test tömege: 1-1 kg,

a két test anyagának fajhője: 4,2-4,2 kJ/(kg °C).

(Ezeket nem tüntetjük fel a táblázatban.)

Független változó: a melegebb test kezdeti hőmérséklete.

A melegebb test kezdeti hőmérséklete	A hidegebb test kezdeti hőmérséklete	A kialakuló közös hőmérséklet
20°C	0°C	10°C
40°C	0°C	20°C
60°C	0°C	30°C
80°C	0°C	40°C
100°C	0°C	50°C

Az elemzés során arra kell összpontosítanunk a tanulók figyelmét, hogy ugyanannak a testnek (az 1 kg tömegű, 0°C hőmérsékletű víznek) más-más mértékben emelte a hőmérsékletét a másik, azonos tömegű és anyagú, különböző hőmérsékletű test. (Először 10°C-ra, majd 20°C-ra, ... végül 50°C-ra emelkedett a 0°C-os víz hőmérséklete.) Általánosítva: minél magasabb a melegebb test hőmérséklete, annál nagyobb mértékben képes a vele kölcsönhatásban levő másik, hidegebb test hőmérsékletét emelni. Más szavakkal: minél magasabb valamely test hőmérséklete, annál nagyobb a **változtató képessége** (melegítő képessége), annál nagyobb az **energiája** (belső energiája), - ha egyébként minden más tényező állandó.

A következőkben azt javasoljuk megvizsgálni, hogy milyen mértékben emelkedik az 1 kg, 0°C-os víz hőmérséklete, ha az egy másik edényben levő, 100°C hőmérsékletű, különböző tömegű vízzel van kölcsönhatásban. A program ismételt futtatása és a grafikonok összehasonlítása után például a következő adatokat kaphatjuk. Allandó a két test anyagának fajhője: 4,2-4,2 kJ/(kg °C), a két test kezdeti hőmérséklete: 100°C és 0°C. (Ezeket nem tüntetjük fel a táblázatban.)

Független változó: a melegebb test tömege.

A melegebb test tömege	A hidegebb test tömege	A kialakuló közös hőmérséklet
0,5 kg	0,5 kg	50°C
1,0 kg	0,5 kg	67°C
1,5 kg	0,5 kg	75°C
2,0 kg	0,5 kg	80°C
2,5 kg	0,5 kg	83°C

A táblázat alapján megállapítható, hogy ugyanannak a testnek (az 1 kg tömegű 0°C hőmérsékletű víznek) más-más mértékben emelte a hőmérsékletét a másik, azonos anyagú és hőmérsékletű, de különböző tömegű test. Minél nagyobb tehát a test tömege, annál nagyobb mértékben képes a vele kölcsönhatásban levő másik, hidegebb test hőmérsékletét emelni. Más szavakkal: minél nagyobb valamely test tömege (azonos hőmérséklet és fajhő esetén), annál nagyobb a **változtató képessége** (melegítő képessége), annál nagyobb az **energiája** (a belső energiája).

Tanulságos azt is megvizsgálni, hogy milyen mértékben emelkedik az 1 kg tömegű, 0°C-os víz hőmérséklete, ha abba különböző anyagú, 1 kg tömegű, 100°C hőmérsékletű testet teszünk. A program ismételt futtatása és a grafikonok elemzése, összehasonlítása után például a következő adatokat kaphatjuk:

Allandó a két test tömege: 1-1 kg,  
a két test kezdeti hőmérséklete: 100°C és 0°C.  
(Ezeket nem tüntetjük fel a táblázatban.)

Független változó: a melegebb test fajhője.

A melegebb test		A hidegebb test fajhője	A kialakuló közös hőmérséklet
anyaga	fajhője		
réz	0,4kJ/(kg·°C)	4,2 kJ/(kg·°C)	9°C
vas	0,46kJ/(kg·°C)	4,2 kJ/(kg·°C)	10°C
alumí- nium	0,9kJ/(kg·°C)	4,2 kJ/(kg·°C)	18°C
alko- hol	2,4kJ/(kg·°C)	4,2 kJ/(kg·°C)	36°C
víz	4,2kJ/(kg·°C)	4,2 kJ/(kg·°C)	50°C

A táblázat adataiból kiolvasható, hogy ugyanannak a testnek (az 1 kg tömegű, 0°C hőmérsékletű víznek) más-más mértékben emelte a hőmérsékletét az azonos hőmérsékletű és tömegű réz-, vas- és alumíniumdarab, alkohol és víz. Ezeknek az anyagoknak más-más a fajhőjük. Minél nagyobb a melegebb test anyagának a fajhője, annál nagyobb mértékben képes a vele kölcsönhatásban levő másik test hőmérsékletét emelni. Más szavakkal: minél nagyobb valamely test anyagának fajhője (azonos tömeg és hőmérséklet esetén), annál nagyobb a **változtató képessége** (melegítő képessége), annál nagyobb az **energiája** (belső energiája).

Általános tapasztalat, hogy a tanulók egy része nehezen tudja kettéválasztani a hőmérséklet és a belső energia fogalmát. Ennek egyik okát abban látjuk, hogy a tanulók által korábban vizsgált hőjelenségek döntő többségében annak a testnek volt nagyobb a belső energiája, amelyeknek magasabb volt a hőmérséklete. (Ez az ismeretszerzés korábbi szakaszában elkerülhetetlen.)

A két fogalom megkülönböztetése szempontjából különösen fontosnak látjuk az utóbbi két összefüggés (a tömegtől és a fajhőtől való függés) vizsgálatát. Mindkét esetben arra láthatnak konkrét példákat a tanulók, hogy azonos hőmérséklet esetén is lehet különböző két test változtató képessége, belső energiája.

Várható, hogy a fenti elemzések elvégzése eredményeképpen mélyebb tartalmat nyer a tanulók tudatában a  $\Delta E = c \cdot m \cdot \Delta T$  összefüggés, s nemcsak egy formális kifejezésnek tartják.

- x -

A program lehetőséget kínál annak érzékeltetésére is, hogy termikus kölcsönhatás közben ugyanannyival nő az egyik test belső energiája, mint amennyivel csökken a másik test belső energiája.

**1. példa:** 1,75 kg tömegű 63°C hőmérsékletű alkoholhoz 0,5 kg tömegű 18°C hőmérsékletű vizet öntenek. Az alkohol fajhője 2,4 kJ/(kg·°C), a víz fajhője 4,2 kJ/(kg·°C).

Hasonlítsd össze az alkohol és a víz belsőenergia-változását!

A kialakuló közös hőmérsékletet számítógéppel határozzuk meg.  $48^{\circ}\text{C}$  lesz az eredmény. Az alkohol hőmérséklete ezek szerint  $15^{\circ}\text{C}$ -kal csökken, a víz hőmérséklete  $30^{\circ}\text{C}$ -kal emelkedik.

Ha kiszámíttatjuk a tanulókkal mindkét folyadék belsőenergia-változását, akkor a következőket kapjuk:

#### Alkohol

$$\begin{aligned} m &= 1,75 \text{ kg} \\ c &= 2,4 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \\ \Delta T &= 15^{\circ}\text{C} \end{aligned}$$

$$\begin{aligned} \Delta E &= c \cdot m \cdot \Delta T = \\ &= 2,4 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \cdot 1,75 \text{ kg} \cdot 15^{\circ}\text{C} = \\ &= 63 \text{ kJ} \end{aligned}$$

#### Víz

$$\begin{aligned} m &= 0,5 \text{ kg} \\ c &= 4,2 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \\ \Delta T &= 30^{\circ}\text{C} \end{aligned}$$

$$\begin{aligned} \Delta E &= c \cdot m \cdot \Delta T = \\ &= 4,2 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \cdot 0,5 \cdot \text{kg} \cdot 30^{\circ}\text{C} = \\ &= 63 \text{ kJ} \end{aligned}$$

Az alkohol belső energiájának a csökkenése tehát egyenlő a víz belső energiájának a növekedésével.

**2. példa:** A 6 kg tömegű,  $22^{\circ}\text{C}$  hőmérsékletű petróleumba egy 4,2 kg tömegű,  $120^{\circ}\text{C}$  hőmérsékletű, acélból készült gépkatrészt tesznek. A petróleum fajhője  $2,1 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C})$ , az acél fajhője  $0,5 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C})$ . Hasonlítsd össze a petróleum és az acél belsőenergia-változását!

A kialakuló közös hőmérsékletet számítógéppel határozzuk meg. Az eredmény  $36^{\circ}\text{C}$  lesz. A petróleum hőmérséklete ezek szerint  $14^{\circ}\text{C}$ -kal emelkedik, az acél hőmérséklete pedig  $84^{\circ}\text{C}$ -kal csökken. Ha kiszámíttatjuk a tanulókkal a petróleum és az acél belsőenergia-változását, akkor a következőket kapjuk:

#### Petróleum

$$\begin{aligned} m &= 6 \text{ kg} \\ c &= 2,1 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \\ \Delta T &= 14^{\circ}\text{C} \\ \Delta E &= c \cdot m \cdot \Delta T = \\ &= 2,1 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \cdot 6 \text{ kg} \cdot 14^{\circ}\text{C} = \\ &= 176,4 \text{ kJ} \end{aligned}$$

#### Acél

$$\begin{aligned} m &= 4,2 \text{ kg} \\ c &= 0,5 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \\ \Delta T &= 84^{\circ}\text{C} \\ \Delta E &= c \cdot m \cdot \Delta T = \\ &= 0,5 \text{ kJ}/(\text{kg}\cdot^{\circ}\text{C}) \cdot 4,2 \text{ kg} \cdot 84^{\circ}\text{C} = \\ &= 176,4 \text{ kJ} \end{aligned}$$

A petróleum belső energiájának a növekedése tehát egyenlő az acél belső energiájának a csökkenésével.

(Megjegyzés: A számítógép egészekre kerekítve adja meg a kialakuló közös hőmérsékletet. Ezért egyes esetekben kis eltérés adódhat a két belsőenergia-változás számított értékei között.)

#### 1.3.1.4. Táblázatok elemzése

A fizikai fogalmak kialakításához, értelmezéséhez, az összefüggések megértéséhez, elmélyítéséhez jól használhatók a táblázatok. A számítógéppel képernyőre vitt táblázatok nagy előnye, hogy gyorsan, egyszerűen módosíthatók az adatok, újabb és újabb kapcsolatok ismertethetők fel.

Könyvünk kétféle táblázat kialakításához tartalmaz programokat a 6-8. osztályban tanított fogalmakkal, összefüggésekkel kapcsolatban. Először a "Táblázat 1." jelzéssel ellátott programok használatával kapcsolatban szeretnénk néhány módszertani javaslatot adni.

#### a/ SŰRŰSÉG (TÁBLAZAT 1.)

A tanulók a 6. osztályban ismerkednek meg a sűrűség fogalmával. Először mérések alapján azt állapítják meg, hogy az azonos anyagú testeknek annyszor nagyobb a térfogatuk, ahányszor nagyobb a tömegük. (Vagy: ahányszor nagyobb a térfogatuk, annyszor nagyobb a tömegük is.) A következő órákon pedig eljutnak annak megállapításáig, hogy az egyenlő térfogatú testek közül a nagyobb tömegű testnek nagyobb a sűrűsége. Végül megismerik a sűrűség kiszámítási módját: A test anyagára jellemző sűrűséget úgy kapjuk meg, hogy a test tömegét elosztjuk a test térfogatával. Hatékony segítséget nyújt a fentiek megvalósításához a SŰRŰSÉG (TÁBLAZAT 1.) című program.

A tanulók mérés alapján összehasonlították az egyenlő térfogatú alumínium-, vas-, réz- és fahenger tömegét. Meghatározták, összehasonlították ezeknek az anyagoknak a sűrűségét is.

E tapasztalatok megerősítését, kibővítését segíti elő a program segítségével megjeleníthető táblázat:

	TÖMEG (g)	TÉRFOGAT (cm <sup>3</sup> )	SŰRŰSÉG (g/cm <sup>3</sup> )
Alkohol	7,9	10	0,79
Alumínium	27,0	10	2,70
Arany	193,2	10	19,32
Gyémánt	35,0	10	3,50
Higany	135,9	10	13,59
Jég 0 fok	9,2	10	0,92
Lítium	5,3	10	0,53
Ólom	113,4	10	11,34
Vas	78,7	10	7,87
Víz 4 fok	10,0	10	1,00

(Megjegyzés: A táblázat néhány olyan anyagnak a sűrűségét is tartalmazza, amely a tankönyv táblázatában nem szerepel. A sűrűség-adatok két tizedesjegy-pontosságig adottak. A táblázat abban a sorrendben tartalmazza az egyes mennyiségeket, ahogy az alapképlet alkalmazásakor a megadott mennyiségekből a végeredményt megkapjuk: tömeg - térfogat - sűrűség.)

A táblázat alapján megállapíthatják a tanulók, hogy például az első három anyag közül az alkoholnak a legkisebb a sűrűsége (0,79 g/cm<sup>3</sup>), ezt követi az alumínium sűrűsége (2,70 g/cm<sup>3</sup>), a legnagyobb az arany sűrűsége (19,32 g/cm<sup>3</sup>). Ha mindhárom anyagból 10-10 cm<sup>3</sup> térfogatú darabot veszünk, akkor azok tömege annál nagyobb (7,9 g, 27,0, 193,2 g), minél nagyobb a sűrűségük. Hasonló megállapításra juthatunk a többi adat összehasonlítása esetén is. Általánosan fogalmazva: az egyenlő térfogatú testek közül annál nagyobb valamely test tömege, minél nagyobb az anyagának a sűrűsége.

Célszerű ezután valamennyi anyag térfogatát egyidejűleg 20 cm<sup>3</sup>-re, majd 50 cm<sup>3</sup>-re, 100 cm<sup>3</sup>-re átírni a géppel. A sűrűség- és tömegadatok összehasonlítása most is az előbbi megállapításhoz vezet.

- x -

A tanulók mérés vagy becslés alapján azt is megállapították, hogy melyik a legnagyobb, illetve a legkisebb térfogatú az egyenlő tömegű alumínium-, vas-, réz- és fahenger közül. E tapasztalatok megerősítését, kibővítését segíthetjük oly módon, ha az első táblázat adatait úgy módosítjuk, hogy mindegyik anyagból egyenlő tömeget veszünk. Ha 10 g-nak választjuk a tömeget, akkor a képernyőn a következő táblázat jelenik meg:

	TÖMEG (g)	TÉRFOGAT (cm <sup>3</sup> )	SŰRŰSÉG (g/cm <sup>3</sup> )
Alkohol	10	12,7	0,79
Alumínium	10	3,7	2,70
Arany	10	0,5	19,32
Gyémánt	10	2,9	3,50
Higany	10	0,7	13,59
Jég 0 fok	10	10,9	0,92
Lítium	10	18,9	0,53
Ólom	10	0,9	11,34
Vas	10	1,3	7,87
Víz 4 fok	10	10,0	1,00

E táblázat alapján most azt állapíthatják meg a tanulók, hogy ha például az első három anyagból 10-10 g tömegű darabot veszünk, akkor azok térfogata annál kisebb (12,7 cm<sup>3</sup>, 3,7 cm<sup>3</sup>, 0,5 cm<sup>3</sup>), minél nagyobb a sűrűségük (0,79 g/cm<sup>3</sup>, 2,70 g/cm<sup>3</sup>, 19,32 g/cm<sup>3</sup>). Hasonló megállapításra juthatunk a többi adat összehasonlítása alapján is. Általánosan: az egyenlő tömegű testek közül annál kisebb valamely test térfogata, minél nagyobb az anyagának a sűrűsége.

Javasoljuk ezután valamennyi anyag tömegét egyidejűleg 20 g-ra, majd 50 g-ra, 100 g-ra átírni a géppel. A sűrűség- és térfogat-adatok összehasonlítása most is az előzővel azonos megállapításra ad alapot.

- x -

A tanulók méréseket végeztek annak megállapítására is, hogy mekkora a különböző nagyságú alumíniumhasábok térfogata és tömege. E tapasztalatok megerősítését, kibővítését is jól szolgálhatja a program.

Fokozatosan növeljük ugyanazon anyag térfogatát, leolvassuk a gép által kiírt tömegadatokat és táblázatba foglaljuk azokat. Az alkohol esetében például a következőket kaphatjuk:

Tömeg	Térfogat
7,9 g	10 cm <sup>3</sup>
15,8 g	20 cm <sup>3</sup>
23,7 g	30 cm <sup>3</sup>
31,6 g	40 cm <sup>3</sup>
39,5 g	50 cm <sup>3</sup>

Megfigyelhetjük a tanulókkal, hogy a  $20 \text{ cm}^3$  térfogatú alkohol tömege 2-szer, a  $30 \text{ cm}^3$  térfogatú alkohol tömege 3-szor akkora, mint a  $10 \text{ cm}^3$  alkoholé. Általánosítva: ahányszor nagyobb az alkohol térfogata, annyiszor nagyobb a tömege is. (Ez az összefüggés megfordítva is igaz: ahányszor nagyobb az alkohol tömege, annyiszor nagyobb a térfogata is.)

A táblázat adataiból az is megállapítható, hogy az alkohol tömegének és térfogatának a hányadosa mindegyik esetben  $0,79 \text{ g/cm}^3$ , ami tulajdonképpen az alkohol sűrűsége. Hasonló megállapításokat tehetünk bármelyik anyagra vonatkozóan is.

- x -

A program jó segítséget nyújt az elsődleges rögzítést szolgáló feladatok megoldásához, ellenőrzéséhez is. Egyszerűvé válik a csoportmunka keretében végzett feladatmegoldás is a program alkalmazásával. Azt a feladatot adjuk például az egyes csoportoknak, hogy számítsák ki a  $15 \text{ cm}^3$  térfogatú testek tömegét. Ehhez mindegyik csoportnak egy-egy olyan anyag sűrűségét adjuk meg, amely szerepel a program táblázatában.

A feladatok megoldása után a táblázatban mindegyik anyaghoz kiíratjuk a géppel a  $15 \text{ cm}^3$ -t, illetve az ennek megfelelő tömeget. A képernyőről minden csoport ellenőrizheti a végzett számítás helyességét.

#### b/ SEBESSÉG (TÁBLAZAT 1.)

A sebesség szót jól ismerik a tanulók a hétköznapi szóhasználatból is. Ennek a szintnek megfelelő módon használják a 6. osztályban a mozgásállapot-változások vizsgálatánál. A 8. osztályban kerül sor arra, hogy a sebességet az egyenletesen mozgó testek mozgását jellemző mennyiségként meghatározzák. Ebben az osztályban tananyag a sebesség kiszámítása az útból és a megtételéhez szükséges időből. A sebesség fogalmának értelmezéséhez, elmélyítéséhez, az út, az idő és a sebesség közötti összefüggés elemzéséhez nyújt segítséget a SEBESSÉG (TÁBLAZAT 1.) című program. A program indítása után a gép tízféle test sebességét írja ki, s megadja mindegyik esetében a 10-10 másodperc alatt megtett utat:

	ÚT (m)	IDŐ (s)	SEBESSÉG (m/s)
Gyalogos	14	10	1,4
Kerékpáros	50	10	5,0
Trabant	300	10	30,0
Helikopter	800	10	80,0
Boeing 707	2800	10	280,0
Hang (levegőben)	3400	10	340,0
TU 144	6900	10	690,0
Hang (vasban)	51000	10	5100,0
Űrhajó	79000	10	7900,0
Föld	298000	10	29800,0

(Megjegyzés: A táblázat több olyan sebességadatot tartalmaz, amely a tankönyvben nem szerepel. Célszerű ezek felhasználásával is bővíteni a megoldandó feladatok témáját.)

Javasoljuk kiindulásképpen néhány sebességadat értelmezését. Az űrhajó sebességének az elképzeltetéséhez javasoljuk megjelölni az iskolától kb. 8 km távolságra levő helyet, s elképzeltetni a tanulókkal, hogy az űrhajó ekkora távolságot tesz meg, miközben 1 másodperc telik el. (Közben metronómmal vagy tapssal jelezzük az 1 másodpercnyi időt.)

A táblázatban növekvő nagyságrendi sorrendben szerepelnek a sebességadatok. Ez könnyen áttekinthetővé teszi azokat. Ha összehasonlíttatjuk a tanulókkal a 10-10 másodperc alatt megtett utakat, jól látszik, hogy minél nagyobb valamely test sebessége, annál nagyobb az azonos idő alatt megtett út is.

- x -

Ezt követően célszerű azt elemezni, hogy milyen összefüggés van a sebesség és az idő között, ha állandó az út. Más megfogalmazásban: azt vizsgáljuk, hogy mennyi időre van szükségük a különböző sebességgel haladó testeknek ugyanakkora út megtételéhez. Ha 1500 m-nek választjuk a testek által megtett utat, akkor a képernyőn a következő táblázatot láthatjuk:

	ÚT (m)	IDŐ (s)	SEBESSÉG (m/s)
Gyalogos	1500	1071,4	1,4
Kerékpáros	1500	300,0	5,0
Trabant	1500	50,0	30,0
Helikopter	1500	18,7	80,0
Boeing 707	1500	5,4	280,0
Hang (levegőben)	1500	4,4	340,0
TU 144	1500	2,2	690,0
Hang (vasban)	1500	0,3	5100,0
Űrhajó	1500	0,2	7900,0
Föld	1500	0,1	29800,0

(Megjegyzés: Célszerű legalább 1500 m-nek venni a testek által megtett utat, mert csak ebben az esetben adódik az utolsó három testre egymástól eltérő időadat. Kisebb utak megadásakor az utolsó két vagy három időadat azonosnak adódik a kerekítés miatt. A tanulók egy részének ez zavaró lehet, mivel úgy tűnhet az adatok alapján, mintha a különböző sebesség ellenére egyenlő idő alatt tennék meg e testek az azonos távolságot.)

Érdekes a tanulók számára, ha a táblázat alapján összehasonlítjuk néhány test konkrét adatpárját. Például a Trabantnak 50 másodpercre (majdnem 1 percre) van szüksége ahhoz, hogy az 1500 m utat megtegye, a Boeing 707-es repülőgép ugyanekkora utat 5,4 másodperc alatt tesz meg. Hasonló összehasonlításokat tehetünk más testek esetében is.

A táblázatból jól kiolvasható, hogy azonos út megtételéhez annál rövidebb időre van szüksége valamely testnek, minél nagyobb a sebessége.

- x -

A program alkalmas arra is, hogy erősítsük annak az összefüggésnek az ismeretét, mely szerint az út és az idő között egyenes arányosság van, ha állandó a sebesség. Ehhez fokozatosan növeljük az ugyanazon testre vonatkozó időadatot, leolvassuk a gép által kiírt utat, majd táblázatba foglaljuk az adatokat.



A helikopterre vonatkozóan például a következőket kaphatjuk:

IDŐ	ÚT
10 s	800 m
20 s	1600 m
30 s	2400 m
40 s	3200 m
50 s	4000 m
60 s	4800 m

Megfigyeltethetjük a tanulókkal, hogy a helikopter 20 s alatt 2-szer, 30 s alatt 3-szor, ... 60 s alatt 6-szor akkora utat tesz meg, mint 10 s alatt. Általánosan: ahányszor hosszabb ideig halad a test, annyiszor nagyobb az általa megtett út. A 8. osztályban már ismert matematikából az arányosság fogalma. Ezért a fenti összefüggést úgy is megfogalmaztathatjuk a tanulókkal, hogy az idő és az út között egyenes arányosság van (egyenletes mozgás).

- x -

Lényegében a fentiekhez hasonlóan alkalmazhatjuk a kapcsolódó tananyag feldolgozásakor a **TELJESÍTMÉNY**, a **HATÁSFOK**, a **SÜRLŐDÁS**, a **HÖTAGULÁS** című programoknak azt a változatát, amelyeknek címe után zárójelben a "Táblázat 1." jelzés olvasható.

#### c/ MUNKA (TÁBLÁZAT 2.)

A tanulók a 6. osztályban ismerkednek meg a munka fogalmának fizikai értelmezésével. Megtanulják, hogy a munka az erő és az erő irányába történő elmozdulás szorzataként számítható ki. A jelenlegi tanterv szerint a munka az első olyan fogalom, amellyel kapcsolatban képletet írnak fel a tanulók és számítási feladatokat oldanak meg. Nagyon fontos, hogy a képlet ismerete, használata ne váljék formálissá. Fontos, hogy minél sokoldalúbban lássák a képlet mögött a tanulók a fizikai tartalmat, a benne szereplő mennyiségek közötti összefüggéseket. Ehhez nyújt lehetőséget a MUNKA (TÁBLÁZAT 2.) című program.

Először azt javasoljuk értelmezni, tudatosítani, hogy milyen összefüggés van az erő irányába eső elmozdulás és a munka között. Ehhez állandónak kell választanunk az erőt, független változónak az elmozdulást, és a munka lesz a függő változó. (Ez a program 1. pontja.) A táblázat egy lehetséges változata a következő:

MUNKA (J)	ERŐ (N)	ELMOZDULÁS (m)
50	10	5
100	10	10
150	10	15
200	10	20
.	.	.
.	.	.
.	.	.
500	10	50

A képernyőn megjelenő táblázatról leolvashatják a tanulók, hogy ugyanakkora erő (10 N) esetén, ha 2-szer, 3-szor, ... 10-szer nagyobb az elmozdulás, akkor 2-szer, 3-szor, ... 10-szer nagyobb a

végzett munka is. Általánosabban fogalmazva: azonos erő esetén ahányszor nagyobb az elmozdulás, annyiszor nagyobb a végzett munka. (Az egyenes arányosságot matematikából csak a 7. osztályban tanulják tanítványaink; így ezt a kifejezést most nem használjuk. Ez az elemzés azonban tényanyagot nyújt az egyenes arányosság tanításához is.)

- x -

A következőkben azt vizsgáljuk, hogy milyen összefüggés van az erő és a végzett munka között, ha állandó az erő irányába eső elmozdulás. (Ez a program 2. pontja.) Egy lehetséges változat a következő:

MUNKA (J)	ERŐ (N)	ELMOZDULÁS (m)
40	8	5
80	16	5
120	24	5
160	32	5
.	.	.
.	.	.
.	.	.
400	80	5

A táblázat alapján megállapíthatják a tanulók, hogy ugyanakkora elmozdulás (5 m) esetén, ha 2-szer, 3-szor, ... 10-szer nagyobb az erő, akkor 2-szer, 3-szor, ... 10-szer nagyobb a végzett munka is. Általánosabb megfogalmazásban: azonos elmozdulás esetén ahányszor nagyobb az erő, annyiszor nagyobb a végzett munka.

- x -

Javasoljuk annak elemzését is, hogy milyen összefüggés van az erő és az erő irányába eső elmozdulás között azonos munka esetén. Ehhez állandónak kell választanunk a munkát, független változónak az elmozdulást, és az erő lesz a függő változó.

(Ez a program 3. pontja.) A táblázat egy lehetséges változata a következő:

MUNKA (J)	ERŐ (N)	ELMOZDULÁS (m)
480	120	4
480	60	8
480	40	12
480	30	16
.	.	.
.	.	.
.	.	.
480	12	40

A táblázatról leolvastathatjuk a tanulókkal, hogy ugyanakkora munka esetén, ha 2-szer, 3-szor, ... 10-szer akkora az elmozdulás, akkor 2-szer, 3-szor, ... 10-szer kisebb az erő. Általános megfogalmazásban: azonos munka esetén ahányszor nagyobb az elmozdulás, annyiszor kisebb az erő.

- x -

A megszokottól eltérően kell végezniük a tanulóknak az elemzést akkor, ha a független változó a második esetben nem a kétszerese

az elsőnek, vagyis a mennyiségek megadásakor nem azonosnak választjuk a kezdő értéket és a lépésközt. Ha például az 1. pontnak megfelelően vizsgáljuk az összefüggéseket, és az elmozdulás kezdő értékét 5 m-nek, a lépésközt 20 m-nek adjuk meg, akkor a képernyőn a következő táblázat jelenik meg:

MUNKA (J)	ERŐ (N)	ELMOZDULÁS (m)
50	10	5
250	10	25
450	10	45
650	10	65
.	.	.
.	.	.
.	.	.
1850	10	185

A megszokott formulától eltérően most úgy kell fogalmazniok a tanulóknak, hogy azonos erő esetén, ha 5-ször, 9-szer, 13-szor nagyobb az út, akkor 5-ször, 9-szer, 13-szor nagyobb a végzett munka is. Az ilyen táblázatok elemzése elejét veheti annak, hogy a tanulók rutinból próbáljanak megfogalmazni valamilyen összefüggést, az egyes mennyiségek összehasonlítása, megfontolt elemzése helyett.

Hasonló táblázat természetesen előállítható a többi összefüggés vizsgálatához is.

- x -

A program további három összehasonlítási lehetőséget tartalmaz attól függően, hogy melyik mennyiséget választjuk állandónak, illetve független és függő változónak (4-6. pont). Ezek felhasználása a fentiekhez hasonlóan történhet.

- x -

Könyvünkben további "Táblázat 2." jelzésű program található **SÜRÜSÉG, NYOMÁS, FORGATÓNYOMATEK, HATASFOK, SEBESSÉG, LENDÜLET, MOZGÁSI ENERGIA, ELLENÁLLAS** címmel. Ezek felhasználása ugyancsak a fentiekhez hasonlóan történhet.

### 1.3.2. Javaslatok az alkalmazáshoz, gyakorláshoz, ellenőrzéshez

A tanulókat "teljesítményképes" tudás birtokába kell juttatnunk. Ez azt jelenti, hogy a tanulóknak nemcsak ismerniök, tudniök kell a tantervi anyagot, hanem képessé kell tennünk őket arra, hogy ismereteiket új szituációkban is tudják használni, alkalmazni. Ehhez viszonylag nagyszámú feladatmegoldásra, gyakorlásra van szükség.

A tanítás-tanulás folyamatában az alkalmazás fázisa szorosan kapcsolódik egyrészt az új ismeretek feldolgozásához, másrészt az ellenőrzéshez. Amikor például az új ismeretek feldolgozásakor megismerkednek a tanulók egy új fizikai összefüggéssel, s az azt kifejező képlettel, akkor annak megértetéséhez szükséges egy-egy mintapélda megoldása. Ez azonban már úgy is tekinthető, mint az alkalmazás első lépése. Amikor viszont a feladatmegoldást gyakoroltatjuk, akkor szükséges, hogy ellenőrizzük a feladatok megoldásának a helyességét. Ez alapfeltétel a visszajelzéshez, a jó megoldások megerősítéséhez, a hibák helyesbítéséhez.

Ennek megfelelően, az alkalmazáshoz, gyakorláshoz kidolgozott programok többsége a tanítási-tanulási folyamat több szakaszában is felhasználható. Kidolgozásukkor azonban az alkalmazás, gyakorlás, ellenőrzés lehetőségének a biztosítását tartottuk elsődlegesen szem előtt.

#### a/ V-A MŰSZER

A tanulóknak kellő jártasságot kell szerezniük - többek között - a feszültség és az áramerősség mérésében. Méréseikhez tanuló-kísérleti műszert használnak. Sok tanuló számára azonban nehézséget jelent a műszerről való leolvasás. Ennek gyakorlásához ad lehetőséget a V-A MŰSZER című program.

A programot alkalmazhatjuk a méréshatár, a skálaleolvasás magyarázatához osztálykeretben is, de elsődlegesen egyéni foglalkozáshoz ajánljuk, olyan tanulók számára, akiknek az átlagosnál nagyobb nehézséget jelent a méréshatár megállapítása, a feszültség és áramerősség leolvasása.

Mivel az iskolákban kétféle tanuló-kísérleti műszer létezik, eltérő skálával és méréshatárokkal, a program mindkét változathoz igazodva ad gyakorlási lehetőséget. A méréshatár megválasztása után a program felrajzolja a skálát, s alatta nyíllal jelöli, hogy a műszer melyik két kivezetéséhez csatlakoztatjuk a vezetékeket. A gép véletlenválasztással jelöli ki a mutató állását. A tanulóknak megfelelő mérőszám és mértékegység írásával kell megadniuk a választ.

A program csak akkor fogadja el a választ helyesnek, ha a mérőszám és a mértékegység is jó. Akkor is hibás a válasz, ha hiányzik a mértékegység.

#### b/ OHM TÖRVÉNYE

A tanulók a 7. osztályban megfelelő számú kísérlet és mérés általánosításaként fogalmazzák meg, hogy ugyanazon fogyasztó kivezetésein mért feszültség és a fogyasztón átfolyó áram erőssége egyenesen arányos, hányadosuk állandó. E törvény megfogalmazása után ismerik meg az ellenállás kiszámításának módját ( $R = U/I$ ), és értelmezik az ellenállás mértékegységét. Ehhez nyújt segítséget az OHM TÖRVÉNYE című program.

A program betöltése, indítása után egy áramkör rajza és egy koordinátarendszer jelenik meg a képernyőn. A rajz alatt a feszültség és az áramerősség adatai olvashatók. (Indulásként:  $U = 0$ ,  $I = 0$ )

A feszültség voltonkénti növelésével egyidejűleg nő az áramerősség is. (Az ellenállás állandó.) Közben fokozatosan rajzolja a gép a grafikont. Célszerű először olyan esetben leállni és értelmezni az adatokat, amikor az áramerősség mérőszáma egész szám, a későbbiek során azonban más adatokat is elemeztethetünk. Egyszerűbbé tehetjük az összetartozó adatpárok áttekintését, ha táblázatba foglaljuk azokat.

Például:

FESZÜLTSEG	ÁRAMERŐSSÉG
8 V	1 A
16 V	2 A
24 V	3 A
28 V	3,5 A
36 V	4,5 A

A táblázathoz kapcsolódóan a következő kérdéseket tehetjük fel a tanulóknak:

- Mit jelent ebben az esetben, hogy: "a fogyasztó kivezetéseinek mért feszültség és a fogyasztón átfolyó áram erőssége egyenesen arányos"?

- Miként kell értelmeznünk ebben az esetben, hogy e két mennyiség hányadosa állandó?

- Mennyi ebben az esetben az összetartozó két-két mennyiség hányadosa?

- x -

A program lehetőséget ad arra is, hogy bármely adatpár esetén kiírassuk az ellenállás értékét. Az ismételt kiíratás - a táblázat elemzésével azonos tartalommal, de más formában - tudatosítja azt az összefüggést, hogy ugyanazon fogyasztó esetében, ha (bizonyos határon belül) változtatjuk a feszültséget és ezzel egyidejűleg változik az áramerősség is; a kettő hányadosa változatlan marad. Olyan megoldást is választhatunk, hogy először a kiírt mennyiség-pár vagy a grafikon alapján meghatározzuk a tanulókkal az ellenállást, majd ellenőrzésként kiíratjuk a géppel.

- x -

Jól szolgálja az ellenállás fogalmának a megértését, elmélyítését, a tanult ismeretek gyakorlását a grafikon elemzése. Megkerestetjük a tanulókkal, hogy a képernyőn látható grafikonon mekkora feszültség tartozik az 1 A áramerősséghez. Az 5 ohmos ellenállás esetén például azt találják, hogy 5 V az 1 amperhez tartozó feszültség. Az 5 ohm ellenállás tehát azt jelenti, hogy ha a fogyasztót 5 V feszültségű áramforráshoz kapcsoljuk, akkor azon 1 A erősségű áram halad át.

Ha két, különböző fogyasztó ellenállását ábrázoló grafikont rajzoltatunk ki a képernyőre, akkor a következő módon hasonlíthatjuk össze azokat:

Megfigyeljük, mekkora feszültség tartozik ugyanazon áramerősséghez mindkét fogyasztó esetében. A különböző feszültségből következtethetünk az ellenállások egymáshoz viszonyított nagyságára. Például azt látjuk, hogy az egyik grafikonon 40 V feszültség, a másik grafikonon 60 V feszültség tartozik az 5 A áramerősséghez. Összehasonlítás az ellenállás értelmezése alapján: annak a fogyasztónak nagyobb az ellenállása, amelyiket a nagyobb feszültségű (60 V) áramforráshoz kell kapcsolnunk ahhoz, hogy ugyanakkora erősségű áram (5 A) haladjon át rajta, mint a másikon a kisebb feszültségű (40 V) áramforráshoz kapcsolva. "Ellenőrzésképpen" célszerű kiszámíttatnunk a két fogyasztó ellenállását, s összehasonlíttatnunk a kapott eredményeket a gép által kiírt adatokkal (12 ohm, 8 ohm).

Másik összehasonlítási lehetőség: megfigyeljük a képernyőn látható két grafikonon, mekkora áramerősség tartozik ugyanazon feszültséghez mindkét fogyasztó esetében. A különböző áramerősségből következtetünk az ellenállások egymáshoz viszonyított nagyságára. Például azt látjuk, hogy az egyik grafikonon 10 A, a másikon 5 A tartozik a 150 V feszültséghez. Összehasonlítás az ellenállás értelmezése alapján: annak a fogyasztónak nagyobb az ellenállása, amelyen keresztül kisebb (5 A) erősségű áram halad át, mint a másikon (10 A), ha mindkettőt ugyanakkora (150 V) feszültségű áramforráshoz kapcsoljuk. Ellenőrzésként most is célszerű kiszámíttatnunk a két fogyasztó ellenállását, s összehasonlíttatnunk a kapott eredményeket a gép által kiírt adatokkal (15 ohm, 30 ohm).

- x -

Jól használható a program az alkalmazás fázisában a feladatmegoldás csoportmunkában történő gyakorlására. Minden csoportnak azt a feladatot adjuk, hogy számítsák ki a számukra megadott mennyiségek alapján a fogyasztó ellenállását. Ezután a számítógéppel egyre nagyobb feszültség- és áramerősség-adatokat íratunk ki, s ezeket adjuk megadott mennyiségenként az egyes csoportoknak, mindegyiknek mást és mást.

Miután a tanulók végeztek az ellenállás kiszámításával, nagyon egyszerű az ellenőrzés: minden csoport ugyanazt az eredményt kell, hogy kapja. A géppel történő kiíratás után a tanulók ellenőrizhetik a megoldás helyességét. Fontos, hogy azt is megállapítsuk (vagy a tanulókkal állapíttassuk meg), hogy hibás eredmény esetén mi okozta a rossz eredményt, s miképpen lehet a jövőben azt elkerülni.

#### *c/ FORGATÓNYOMATÉK. (GYAKORLÁS 1.)*

A forgatónyomaték a 7. osztályban tananyag. A tanulók a kétoldalú emelőn végzett kísérletsorozat alapján ismerik fel, hogy az erő forgató hatása, a forgatónyomaték az erő nagyságától és az erőkartól függ. Megtanulják a forgatónyomaték kiszámítási módját (erő erőkar), s azt az egyszerű gépekkel kapcsolatos feladatok megoldásában alkalmazzák. Ehhez nyújt további gyakorlási lehetőséget a FORGATÓNYOMATÉK (GYAKORLÁS 1) című program.

Indításkor kilenc pontos "menüt" ajánl a program. Az első három pont olyan szöveg nélküli számítási feladatok megoldását kívánja meg a tanulóktól, amelyekhez előre meghatározhatjuk, hogy melyik legyen az összefüggésben szereplő mennyiség (forgatónyomaték, erőkar, erő) közül a kiszámítandó mennyiség.

A tapasztalatok azt mutatják, hogy a tanulóknak egyszerűbb, könnyebb a szöveg nélküli feladatok megoldása, mint a szövegeseké. Ezért azt javasoljuk, hogy a program alkalmazását ezzel kezdjük. Az első három pontban kínált feladatok megoldása hatékony segítséget nyújt a megoldás algoritmusának az elsajátításához. E feladattípusok valamelyikét adhatjuk azoknak a tanulóknak, akik ismételten valamilyen típushibát követnek el, például hibásan fejezik ki az alapképletből a keresett mennyiséget (2., 3. pont). E feladattípusok előnye, hogy a tanulóknak ugyanazt az összefüggést kell alkalmazniuk a feladatok megoldásához mindaddig, míg át nem térnek a program másik pontjára. Ez lehetővé teszi számukra a kellő gyakorlottság megszerzését. Egy bizonyos határon

túl azonban mechanikussá válhat az alkalmazás. Ennek megelőzésére célszerű kellő időben áttérni a program 4. pontjának az alkalmazására (vegyes feladatok).

A tanulók e programpont alapján vegyesen, véletlen választás alapján kapják az összefüggésben szereplő három mennyiség valamelyikének a kiszámítását. Ezek megoldásához már minden egyes feladat esetében újra és újra mérlegelniük kell a tanulóknak, hogy a megoldáshoz az alapképletet kell-e alkalmazniuk, vagy előzőleg át kell-e azt alakítaniuk. Ennek megfelelően kell dönteniük a további lépésekről, s megoldaniuk a feladatot.

- x -

Lényegében hasonló felépítésű a program 5-8. pontja is, de az itt szereplő feladatok szöveges feladatok.

- x -

A program csak akkor fogadja el helyesnek a megoldást, ha jó a végeredményben a mérőszám és a mértékegység is. Hibás megoldás esetén jelzi a gép, hogy számítási hibát követett el a tanuló, vagy a mértékegység kiírása volt helytelen. A feladat ismételt hibás megoldása esetén megadja a program a helyes eredményt.

A program 9. pontja lehetőséget nyújt a feladatmegoldásban elért eredmények számbavételére. A gép feladattípusonként és összegezve is megadja az eredményeket (százalékban kifejezve). Ez a visszajelzés jó támpontot ad a további gyakorlás témájának és nehézségi szintjének a megállapításához, vagy az elért jó teljesítmény elismeréséhez.

- x -

A könyvben további programok találhatóak "Gyakorlás 1." jelzéssel a következő címekkel: **MUNKA**, **SŰRŰSÉG**, **SEBESSÉG**, és **ELLENÁLLÁS**. Ezek felhasználását is a fentiekhez hasonlóan javasoljuk.

#### d/ **SEBESSÉG (GYAKORLÁS 2.)**

A sebesség fogalmának értelmezéséhez, elmélyítéséhez jó módszer lehet, ha két-két mennyiség összehasonlítása alapján meghatározzuk a harmadik mennyiséggel fennálló kapcsolatot. A **SEBESSÉG (GYAKORLÁS 2.)** című program ehhez változatos gyakorlási lehetőséget ad, három nehézségi fokozatban (könnyű, közepes, nehéz).

Az első (könnyű) fokozatban olyan feladatok vannak, amelyekben konkrét mennyiségek (8 m, 4 m, stb.) összehasonlítása alapján kell a harmadik adatpárt összehasonlítani. A második (közepes) fokozatban az egyes mennyiségek jeleivel és ezek arányával adottak a feladat feltételei (2s, s, stb.). A harmadik (nehéz) fokozatban ugyancsak a mennyiségek jelei adottak, de nem ismert azok aránya, csak annyi, hogy melyik mennyiség kisebb, nagyobb, vagy egyenlő-e a két mennyiség ( $s_1 > s_2$ , stb.).

Mindhárom nehézségi fokozaton belül megválasztható, hogy melyik legyen az összehasonlítandó mennyiség: az út, az idő vagy a sebesség (1-3. pont). Választható olyan lehetőség is, hogy a program felváltva, vegyesen adja az összehasonlítandó mennyiségeket (4. pont).

Célszerű az egyes feladattípusok megoldatása előtt olyan feladatokat elemeztetnünk, amelyek konkrét gyakorlati példákhoz kapcsolódnak. Így könnyebb elképzelniük, felismerniük a tanulóknak a mennyiségek közötti összefüggéseket. A későbbiek során azonban

szükséges elvontabb formában is megoldatni ilyen feladatokat, éppen a gondolkodás fejlesztése érdekében.

Az első (könnyű) fokozatú, a sebesség összehasonlítását kívánó feladatok megoldatása előtt például a következő (vagy ehhez hasonló) feladat elemzését javasoljuk:

Két autó közül az egyik 60 km utat, a másik 120 km utat tesz meg 1,5 óra alatt. Hasonlítsd össze a két autó sebességét!

Ha e feladatból kiemeljük a fizikai mennyiségeket, elvonatkoztatjuk azokat a konkrét szituációtól, akkor a feltételeket és a megoldást a következő módon írhatjuk fel röviden:

$$\begin{array}{rcl}
 60 \text{ km} < 120 \text{ km} & \text{vagy:} & 60 \text{ km} \quad 120 \text{ km} \\
 \hline
 1,5 \text{ h} = 1,5 \text{ h} & & \hline
 v_1 < v_2 & & 40 \text{ km/h} < 80 \text{ km/h}
 \end{array}$$

A két autó közül tehát - mindkét megoldás szerint - a másodiknak nagyobb a sebessége.

Az első megoldáshoz a sebességnek azt az értelmezését vettük alapul, hogy annak a testnek nagyobb a sebessége, amely ugyanannyi idő alatt nagyobb utat tesz meg. A második megoldáshoz a sebesség kiszámítására tanult összefüggést ( $v = s/t$ ) használtuk fel, anélkül, hogy a két idő- és útadatot egymással összehasonlítottuk volna. E megoldás alapján nemcsak azt tudjuk mondani, hogy a második autó sebessége nagyobb, mint az elsőé, hanem azt is, hogy a második autó sebessége 2-szer akkora, mint az elsőé.

Az alábbiakban sorra vesszünk néhány olyan feladatot, amelyet a program ad a sebesség összehasonlítására a könnyű fokozatban:

$$\begin{array}{rcl}
 4 \text{ m} \quad 2 \text{ m} & & 5 \text{ m} \quad 5 \text{ m} & & 7 \text{ m} \quad 1 \text{ m} \\
 \hline
 8 \text{ s} \quad 8 \text{ s} & & \hline
 v_1 & & v_2 & & v_1 & & v_2
 \end{array}$$

Az első feladat megoldására kétféle lehetőség kínálkozik, a fenti, autóra vonatkozó példa kétféle megoldásával azonos gondolatmenettel.

A második feladat megoldásához a sebességnek azt az értelmezését vesszük alapul, hogy annak a testnek a sebessége nagyobb, amely ugyanazt az utat rövidebb idő alatt teszi meg ( $v_1 < v_2$ ).

Megoldható a feladat a két sebesség konkrét kiszámítása útján is ( $1 \text{ m/s} < 2,5 \text{ m/s}$ ).

A harmadik feladat megoldásához az vezet el, ha meggondoljuk, hogy az első test nagyobb utat (7 m-t) tett meg rövidebb idő (2 s) alatt, mint a másik test hosszabb idő (5 s) alatt. Így az első test sebessége nagyobb ( $v_1 > v_2$ ).

Eljuthatunk e válaszhoz úgy is, ha feltételezzük, hogy a második test változatlan sebességgel halad tovább, s 7 m utat tesz meg az is, akárcsak az első:



$$7 \text{ m} \quad 7 \text{ m}$$

$$\underline{2 \text{ s}} \quad \underline{35 \text{ s}}$$

$$v_1 > v_2$$

Ebben az esetben a megoldás módja már ugyanaz lehet, mint a második feladat esetében.

Végül - természetesen - e feladatot is megoldhatjuk úgy, hogy kiszámítjuk mindkét test sebességét, s e konkrét mennyiségeket hasonlítjuk össze ( $3,5 \text{ m/s} > 0,2 \text{ m/s}$ ).

- x -

A sebesség összehasonlítását igénylő, második (közepes) nehézségi fokozatú feladatok megoldatása előtt a következő (vagy ehhez hasonló) feladat elemzését javasoljuk:

Két autó közül az egyik 2-szer akkora utat tesz meg, mint a másik, ugyanannyi idő alatt. Hasonlítsd össze a két autó sebességét!

A feladatban szereplő mennyiségeket, a köztük levő arányt a megoldással együtt a következő módon írhatjuk fel röviden, az egyes mennyiségek betűjeleinek az alkalmazásával:

$$s < 2 s$$

$$\underline{t} = \underline{t}$$

$$v_1 < v_2$$

A két autó közül tehát a második autónak nagyobb a sebessége. A feladat megoldásához ismételten a sebesség értelmezése ad lehetőséget (különböző út - egyenlő idő).

A következőkben azt vesszük sorra, hogy milyen lehetőségek adódnak a sebesség összehasonlítására a program második (közepes) fokozatából vett feladatok megoldásakor. Három feladat a programból:

$\underline{6 \text{ s}}$	$\underline{2 \text{ s}}$	$\underline{2 \text{ s}}$	$\underline{2 \text{ s}}$	$\underline{3 \text{ s}}$
$\underline{t}$	$\underline{t}$	$\underline{2 t}$	$\underline{3 t}$	$\underline{2 t}$ $\underline{4 t}$
$\underline{v_1}$	$\underline{v_2}$	$\underline{v_1}$	$\underline{v_2}$	$\underline{v_1}$ $\underline{v_2}$

E feladatok során már nem konkrét mennyiségeket kell összehasonlítani a tanulóknak, hanem betűjelekkel megadott mennyiségeket. Ez a tanulók egy része számára problémát jelent. Segítség adhat e tanulók számára az a módszertani megoldás, hogy konkrét mennyiségeket helyettesítettünk a betűjelek helyére, s így végeztetjük el az összehasonlítást, s ezután térünk újra vissza az eredeti feladathoz.

Az első és a második feladat megoldására most is a sebesség két-féle értelmezése ad lehetőséget (különböző út - egyenlő idő, illetve egyenlő út - különböző idő).

A harmadik feladat megoldása azért bizonyul nehéznek sok tanuló számára, mivel nincs benne közvetlenül összehasonlítható két mennyiség. E feladat megoldásához is a konkrét adatok behelyettesítése adhat segítséget.

Másik megoldási mód, hogy kiszámítjuk a sebességnek megfelelően, az út és az idő hányadosát ( $1 \text{ s/t} > 0,75 \text{ s/t}$ .) Ez a megoldási mód természetesen alkalmazható az első és a második feladat esetében is.

(Megjegyzés: Hívjuk fel a tanulók figyelmét arra, hogy az első fokozatú feladatokban az  $s$  betű a másodperc rövidítését jelentette, itt viszont az út betűjeleként szerepel.)

- x -

A harmadik (nehéz fokozatú,  $s$  ezen belül a sebesség összehasonlítását igénylő feladatok megoldása előtt javasoljuk a következő (vagy ehhez hasonló) feladat elemzését:

Két autó közül az egyik nagyobb utat tesz meg, mint a másik, ugyanannyi idő alatt. Hasonlítsd össze a két autó sebességét! A feltételeket és a feladat megoldását a következő módon írhatjuk fel röviden:

$$\begin{aligned} s_1 &< s_2 \\ t_1 &= t_2 \end{aligned}$$

$$v_1 < v_2$$

A két autó közül tehát a második autónak nagyobb a sebessége.

A feladat megoldásához ez esetben is a sebesség értelmezése adta a lehetőséget (különböző út - egyenlő idő).

Az alábbiakban azt vesszük sorra, hogy milyen lehetőségek adódnak a sebesség összehasonlítására a program harmadik (nehéz) fokozatú feladatainak a megoldásához. Négy feladat a programból:

$s_1 < s_2$	$s_1 = s_2$	$s_1 > s_2$	$s_1 > s_2$
$t_1 = t_2$	$t_1 < t_2$	$t_1 < t_2$	$t_1 > t_2$
$v_1 < v_2$	$v_1 < v_2$	$v_1 < v_2$	$v_1 < v_2$

E feladatok megoldásakor is betűjelek alapján kell végezniük a tanulóknak az összehasonlítást. A korábbi feladatokhoz képest nehezíti a feladatok megoldását, hogy most nem ismert az összehasonlítandó két-két mennyiség aránya, nem tudjuk, hogy az egyik test által megtett út hányszorosa a másiknak, vagy az út megtételéhez szükséges idő hányszorosa a másiknak.

Az első és a második feladat megoldásához ismételten a sebesség kétféle értelmezése adja a kulcsot (különböző út - egyenlő idő, illetve egyenlő út - különböző idő).

A harmadik feladat megoldásához az a megfontolás vezethet el, hogy az első test nagyobb utat tett meg rövidebb idő alatt, mint a második, tehát az első testnek nagyobb a sebessége.

A negyedik feladat megoldása különösen tanulságos, ha végig vesszük valamennyi lehetőséget. Az első test sebessége ugyanis lehet nagyobb vagy kisebb, mint a másodiké, vagy lehet a két sebesség egyenlő is, - attól függően, hogy hányszorosa az egyik út a másik útnak, illetve hányszor akkora az egyik idő, mint a másik. Ha probléma merülne fel a feladat megoldásában, akkor célszerű konkrét arányokat (vagy szükség esetén konkrét mennyiségeket) beírni, s úgy megoldani a feladatot, majd ezután visszatérni az eredeti feltételekhez.

- x -

Hasonló lehetőségek kínálkoznak az út és az idő összehasonlítására is, a másik két-két mennyiség alapján. (Mi most csak a sebesség összehasonlítására szolgáló feladatokból idéztünk néhányat.)

- x -

Ez a program is tartalmaz olyan részt (4. pont), amelynek segítségével vegyesen, váltakozva adja a gép az út, az idő és a sebesség összehasonlításához a feladatokat.

Célszerűnek tartjuk valamennyi feladattípus esetében vissza-visszatérni a szöveges értelmezéshez, annak logikai megfordításaként, amit az egyes nehézségi fokozatok bevezetőjeként javasoltunk ("autós" példák). Az utolsó feladathoz például ilyen megfogalmazást adhatnak a tanulók: Jóska nagyobb utat tett meg, mint Béla. Jóska hosszabb ideig volt úton, mint Béla. Hasonlítsuk össze Jóska és Béla sebességét!

- x -

A program lehetőséget nyújt arra, hogy számbavegyük a tanuló által elért eredményeket, külön-külön értékelve az egyes nehézségi fokozatokban, s ezen belül az út, az idő és a sebesség összehasonlításában elért teljesítményeket (százalékban kifejezve).

- x -

További programokat közlünk könyvünkben "Gyakorlás 2." jelzéssel **MUNKA, SÜRÜSÉG, SEBESSÉG, és OHM TÖRVÉNYE** címmel. Ezek felhasználását is a fentiekhez hasonló módon javasoljuk.

#### e/ KAPCSOLÁSOK

A fogyasztók soros és párhuzamos kapcsolása a 8. osztályban tananyag. Tanári és tanulói kísérletek, mérések tapasztalatainak az elemzése után jutnak el a tanulók annak megismeréséig, hogy a sorba kapcsolt fogyasztók ellenállásai összeadódnak, a párhuzamosan kapcsolt fogyasztók eredő ellenállása pedig kisebb, mint az összekapcsolt fogyasztók bármelyikének az ellenállása. (A párhuzamosan kapcsolt fogyasztók eredő ellenállásának a kiszámítása nem tantervi követelmény az általános iskolában.) A KAPCSOLÁSOK című program a sorosan kapcsolt fogyasztók eredő ellenállásának kiszámításához és a párhuzamosan kapcsolt fogyasztók eredő ellenállásának nagyságrendi megállapításához ad gyakorlási és ellenőrzési lehetőséget.

A program indítása után egy kapcsolási rajz és öt vagy hét válasz jelenik meg a képernyőn. A tanulóknak a kapcsolási rajz elemzése és a megadott két ellenállásadat alapján kell meghatározniuk az eredő ellenállást és kiválasztaniuk a helyes választ.

A program négyféle rajzot tartalmaz a fogyasztók soros kapcsolására, és négyféle módon ábrázolja a párhuzamosan kapcsolt fogyasztókat is. Ezek közül váltogatva, véletlenválasztással adja meg a gép valamelyik rajzot. A soros, illetve a párhuzamos kapcsolat felismerése az első feltétel a helyes válaszhoz való eljutáshoz.

A fogyasztók kapcsolási módjának ismeretében kell továbbhaladniuk a tanulóknak a megoldásban. Soros kapcsolat esetén mennyiségileg is meg kell adniuk az eredő ellenállást (a két ellenállás összegeként). Párhuzamos kapcsolásnál viszont azt kell felismerniük és konkretizálniuk az adott esetre, hogy az eredő ellenállás kisebb, mint az összekapcsolt ellenállások bármelyike.

Ezek után, ezek alapján tudják a tanulók biztonságosan (és nem találgatva) kiválasztani a helyes feleletet.

(Megjegyzés: A program hét választ ad meg abban az esetben, ha különböző nagyságú a megadott két ellenállás, és ötöt akkor, ha egyenlő nagyságú a két ellenállás. E válaszok tartalmazzák a helyes, és a tanulók részéről várható hibás válaszok mindegyikét.)

A program a válaszadás után, feladatonként azonnal jelzi a képernyő alján a helyes és a rossz válaszok számát. Hibás válasz esetén villog a jó válasz előtti betű. A gép méri a feladatmegoldásra fordított "tisztá" időt. (Válaszadás után leáll az óra, s csak akkor indul meg újra, ha megjelenik a képernyőn az új feladat.) Az időmérés érdekes lehet a tanulók számára, de az a javaslatunk, hogy a megoldás idejét ne vegyük tekintetbe a tanulók munkájának az értékeléséhez.

#### 1.4. A programok felhasználási lehetőségei az 1978-as tanterv megvalósításához

A következőkben arról adunk táblázatos áttekintést, hogy az 1978-ban bevezetett általános iskolai fizika tanterv mely témaköréhez, s azon belül melyik tananyagrészhöz tartalmaz könyvünk programot. A tantervi anyag részletes kifejtését a tankönyvek tartalmazzák, s a tananyag feldolgozása is alapvetően a tankönyvek alapján történik. Ezért a tananyagot az 1983-85-ben átdolgozott 6-8. osztályos munkatankönyvek tananyag-beosztása, fejezetcímei szerint soroljuk fel. Csak azok a fejezetcímek szerepelnek a táblázatban, amelyeknek feldolgozásához program kapcsolódik.

Az egyes fejezetek után olyan csoportosításban közöljük a felhasználásra ajánlott programokat, hogy az minél konkrétabb tájékoztatást adjon az adott program funkciójáról, szerkezetéről. Ennek megfelelően, külön-külön oszlopban tüntetjük fel a kísérletek bemutatását, a mérési eredmények pontosabbá tételét szolgáló, illetve a szimulációs programokat.

Az összefüggések vizsgálatához felhasználásukat tekintve jól különváló, és szerkezetében is különböző két programsorozat szerepel a könyvben. Ezeket célszerűnek tartjuk külön feltüntetni ("Táblázat 1.", "Táblázat 2."). Ugyancsak két, jól elkülöníthető programsorozatot alkot a két-két mennyiség összehasonlítását szolgáló és a feladatmegoldást segítő programsorozat ("Gyakorlás 1.", "Gyakorlás 2."). Ezeket is praktikusnak látszott külön-külön feltüntetni. Végül "Egyéb" címszó alatt szerepelnek azok a gyakorlást szolgáló programok, amelyek egyik csoportba sem sorolhatók.

Ezek szerint a következő csoportosításban tüntetjük fel az egyes tankönyvi fejezetek mellett (rövidítve az egyes elnevezéseket):

- a/ Kísérlet
- b/ Mérés
- c/ Szimuláció
- d/ Táblázat 1.
- e/ Táblázat 2.
- f/ Gyakorlás 1.
- g/ Gyakorlás 2.
- h/ Egyéb

Az a-e/ pont alatt szereplő programok elsődlegesen az új ismeretek feldolgozásához alkalmazhatók, míg a f-h/ pontban feltüntetett programok főleg az alkalmazás, gyakorlás, ellenőrzés során használhatók. Amint erről már korábban szoltunk, ez a csoportosítás csak a programok felhasználásának fő területét jelzi, többségük alkalmazható a tanítás-tanulás más fázisaiban is. Az egyes oszlopokban, a tankönyvi fejezetek sorában a kapcsolódó programokat azokkal a sorszámokkal jelezzük, amelyek alatt könyvünk 2. fejezetében az adott program leírása és listája megtalálható. Minden programot annál a tankönyvi fejezetnél tüntetünk fel, amelynek feldolgozásakor első ízben alkalmazható a program.

Ha valamenyik programhoz, mint "típusprogramhoz" részletes módszertani javaslatok is olvashatók a könyv 1. fejezetében, akkor e tényre bekeretezett sorszám hívja fel a figyelmet a táblázatban.

Tankönyvi fejezet	P r o g r a m							
	a/ K í s é r l e t	b/ M é r é s	c/ S z í m u l á c i ó	d/ T á b l á z a t 1.	e/ T á b l á z a t 2.	f/ G y a k o r l á s 1.	g/ G y a k o r l á s 2.	h/ E g y é b
<u>6. osztály</u>  I. KÖLCSÖNHATÁS, ERŐ, MOZGÁS A testek hőmérséklet- változása..... II. AZ ENERGIA, A MUNKA ÉS A HŐ A munka..... A sűrűség..... A hőmennyiség..... III. HŐJELENSÉGEK A szilárd testek hő- tágulása..... A folyadékok és gázok hőtágulása.....			30					
				3	21 4	19 1	20 2	
		31						
			32					
			33					



Tankönyvi fejezet	P r o g r a m							
	a/ K í s é r l e t	b/ M é r é s	c/ S z i m u l á c i ó	d/ T á b l á z a t  1.	e/ T á b l á z a t  2.	f/ G y a k o r l á s  1.	g/ G y a k o r l á s  2.	h/ E g y é b
<u>8. osztály</u>								
<b>I. A TESTEK MOZGÁSA</b>								
A sebesség.....	46		5	8	9	6	7	
Az átlagsebesség.....			10					
Az egyenletesen változó mozgás.....			11					
			12					
A lendület. A lendület megmaradásának törvénye.			16		17			
A lendület és a mozgási energia.....					18			
Szabadesés, közegellen- állás.....		13						
A súrlódás.....				25				
Rezgőmozgás.....	15		14					
<b>II. AZ ELEKTROMOS ÁRAM HATÁSAI, AZ INDUKCIÓ</b>								
Az elektromos mező munká- ja és a fogyasztók teljesítménye.....					44			
<b>III. FÉNYJELENSÉGEK</b>								
A fényvisszaverődés, a síktükör.....			34					
Gömbtükrök. A domború gömbtükrök képalkotása..			35					
Fénytani lencsék.....			36					
A testek színe.....	37							

## 2. PROGRAMOK

"Tökéletes program nincs, csak még nem találták meg a hibát benne!" (Idézet az ÖTLET című hetilap 1983. decemberi BIT-LET mellékletéből.)

### 2.1. Általános tudnivalók

Az egyes programok kezelésének részletes ismertetése és a programlisták előtt szeretnénk összefoglalni egyrészt azokat a szempontokat, amelyeket a programok kidolgozásánál figyelembe vettünk, másrészt azokat a tudnivalókat, melyeket a programok használatához fontosnak tartunk.

#### 2.1.1. A programok megírásának szempontjai. A programok futtatása

A könyvben található valamennyi program legfontosabb jellemzője, hogy a Commodore 16 számítógép alapváltozatán, memóriabővítő nélkül is használhatók. A programírás egyik fő szempontja volt ez a feltétel, mely esetenként kisebb-nagyobb kompromisszumra is kényszerített. Úgy véljük, hogy most, amikor az iskoláknak csak kis hányada rendelkezik memóriabővítővel, vagy Commodore Plus/4 számítógéppel, helyesebb egy-egy ismertetőt rövidebbre fogni vagy esetenként elhagyni, hogy a program elférjen a C-16 közismerten kis memóriájában. Így ugyanis a Commodore 16-tal rendelkező iskolák mindegyike használhatja az összes programot, a nagyobb memóriájú gépekhez pedig könnyű kiegészíteni, továbbfejleszteni azokat.

Bár az iskolák döntő többsége a magyar ékezetes betűk megjelenítésére alkalmas gépekkel rendelkezik, és az iskolai alkalmazásnál mi is fontosnak tartjuk a helyesírási szempontok figyelembe vételét, a programlistákat mégis az eredeti, az ékezetes karaktereket nem "ismerő" gépre írt változatban közöljük. Elsősorban az előbb is említett ok miatt, azaz valamennyi gépen átírás nélkül is futtatható programokat akarunk közreadni. Másrészt a programlisták elkészítése is lényegesen körülményesebb lett volna, ha a szövegek ékezetes karaktereket is tartalmaznak. A nyomtatók ugyanis az eredeti (grafikus) jeleket nyomtatják az ékezetes betűk helyett.

Ehhez kapcsolódik, hogy a programírásnál ügyeltünk arra, hogy a feliratok az ún. írógép üzemmódban jelenjenek meg, mert csak ebben az üzemmódban van lehetőség az ékezetes betűk megjelenítésére. Ezáltal a magyar ABC megjelenítésére alkalmas gépeken egyszerűen átírhatók a program futása közben kiíródó szövegek. Az írógép üzemmódra történő átkapcsolásról maga a program gondoskodik, és a szokott módon nem is lehet a nagybetűket és grafikus jeleket használó üzemmódhoz visszatérni. Ehhez előbb a PRINT CHR\$(9) parancsot kell beírni és le kell nyomni a RETURN billentyűt. Ezzel elkerülhető, hogy a tanuló véletlen gombnyomással értelmetlen jelműt "varázsoljon" a képernyőre.

Az írógép üzemmód használatának másik előnye, hogy ilyenkor a mértékegységek is helyesen írhatók fel.

A mértékegységek és a képletek szabályos megjelenítésének igénye miatt egyes programokban szükség volt speciális karakterek létrehozására. A kezelési leírásban csak akkor tüntettük fel



ezeket, ha azok valóban a billentyűzet segítségével megjeleníthető karakterek és nem a nagyfelbontású grafikával megrajzolt jelek (pl.  $\Delta$ ,  $\pi$ ,  $\bar{v}$ ) voltak. Az előbbi esetben ugyanis a program leállítása után is a megfelelő billentyűk megnyomására az adott speciális jelet írja ki a gép.

Másik következménye is van a karakterek átdefiniálásának. Egy esetleges hiba elkövetésekor a képernyő teljesen olvashatatlaná válik, "mákos" lesz. Hogy ezt egyszerűen megszüntethessük, az F1 funkcióbillentyűt ezeknél a programoknál úgy programoztuk, hogy megnyomásával visszaállíthassuk a képernyő normális működését, és a gépben lévő program se vesszen el. Ha az eredeti karakterkészletet akarjuk használni a RUN/STOP billentyűt lenyomva tartva nyomjuk meg egy pillanatra a RESET gombot. Ennek hatására a gép ún. monitor üzembe megy, ahonnan az X majd ezt követően a RETURN megnyomásával térnetünk vissza a BASIC-hez. Ennek a kissé bonyolult eljárásnak előnye, hogy a program nem törlődik. Ha a programra nincs szükségünk, a RESET gomb is visszaállítja az eredeti karakterkészletet.

A programíráskor figyelembe vett szempont volt az is, hogy csak olyan színeket használjunk, amelyek a fekete-fehér televízió is megkülönböztethetők. Ettől csupán egyetlen esetben, a SZÍNEK programnál kellett eltérni. Mivel itt éppen a megjelenő különféle színek tanulmányozása a cél, a program alkalmazásához színes televízió szükséges. Minden más program fekete-fehér képernyővel is használható.

A programok közül csak azok használnak hangot, melyeknél a működéshez elengedhetetlen (HANGGENERÁTOR és METRONÓM). Tanórán történő alkalmazásnál ugyanis rendkívül zavaró lehet az esetleg több számítógépből áradó zaj. Ezenkívül feleslegesnek tartjuk az egy-egy test mozgását, egy grafikon rajzolását kísérő hanghatásokat, ha azoknak semmilyen pedagógiai funkciójuk sincs, mert egyrészt elterelik a tanulók figyelmét a lényeges dolgokról, másrészt többnyire lassítják a programok futását.

A programok mindegyike BASIC nyelvű, így a beírás után a RUN paranccsal valamennyi futtatható. A beírt programokat a szokásos SAVE paranccsal kazettára vagy lemezre is kimenthetjük. A kazettáról való betöltés idejét minden egyes program ismertetőjében megadjuk, mivel az a tanórai alkalmazásnál fontos információ lehet. A megadott időtartamok a betöltést gyorsító különféle segédprogramok alkalmazása nélkül készített felvételekre vonatkoznak. Az említett segédprogramok használatával a betöltési idő lényegesen lerövidíthető, de ez esetenként a betöltés megbízhatóságának csökkenésével jár. (A NOVOTRADE egyébként forgalmaz ilyen, ún. TURBO programokat.)

A programok döntő többsége az elindítás után egy rövid ismertetőt ír ki a képernyőre. Ez azonban csak a program kezelésével kapcsolatos legfontosabb információkat tartalmazza. Ezt elolvasva általában egy adat beírása vagy a T billentyű lenyomása után térhetünk át a program "érdemi" részére.

A HELP billentyűnek (help = segítség) általában két funkciója van a programban. Ha egy program futása során egy feladatra adandó válaszra vár a gép, a HELP lenyomásával megkérdezhetjük a helyes választ. Természetesen ekkor ez a "megoldás" nem számít be az eredménybe. Más esetben a HELP gomb lenyomása - segítségként - a program elején található ismertető megjelenítését eredményezi.

Több programban jut szerephez a **CLEAR/HOME** billentyű (clear = tisztítás, home = haza, otthon) is. Önmagában megnyomva valamilyen alaphelyzet beállítására használjuk. Például a **HANGGENERATOR** programban a frekvenciákat a 440 Hz-es alaphelyzetbe állítja, míg a **SÜRÜSÉG** (tábl.1.) programban a jelet viszi a kiindulási helyzetbe (haza), az első sor első oszlopába. A **SHIFT** billentyűvel (shift = váltás) együtt lenyomva valamilyen tisztítási, törlési műveletet végez a programok többsége. Több programban például a gombok megnyomásával törölhetők a koordináta-rendszerbe addig berajzolt grafikonok.

Kiemelt szerepe van természetesen a **RETURN** billentyűnek. A return visszatérést jelent (az új sor elejére). A gépen általában ezzel a billentyűvel kell lezárni minden parancsot, adatbevitelt. Ha a programokban egyetlen karakter hosszúságú választ vár a program, a **RETURN** billentyűt nem kell használni. Ez a helyzet fordul elő például az ismertetők végén, a **TOVABB = T** felirat megjelenésekor, vagy a különböző menükről történő választáskor. Ha azonban a program több karakterből álló választ vár (például egy feladat eredményét), akkor a **RETURN** lenyomásával kell jelezni a programnak, hogy nem akarunk mást beírni. Ha a beírás során valamit elrontottunk, az az **INST/DEL** gomb (insert = beilleszt, delete = töröl) segítségével általában törölhető.

A programok a **RUN/STOP** billentyűvel állíthatók le. A (run = futás, stop = megállás). Ezután a program még a gépben van és újra indítható vagy listázható. Ha a programot olyankor állítottuk meg, amikor az a nagyfelbontású grafikát használta, akkor a **READY** üzenet (ready = kész) és a villogó kurzor nem jelenik meg. Ilyen esetben a **GRAPHIC 0** parancs beírásával és a **RETURN** lenyomásával térhetünk vissza a normál, szöveges üzemmóddhoz. A **GRAPHIC 0** parancsot "vakon" kell beírni, mert a lenyomott gomboknak megfelelő betűk sem látszanak a képernyőn.

A **RESET** gomb (reset = visszaállítás) segítségével az egész számítógép alaphelyzetbe állítható, de használatával vigyázni kell, mert megnyomása egyúttal a programot is törli a memóriából. Előnye, hogy egy gombnyomással mindent alaphelyzetbe állít (színek, funkcióbillentyűk, karakterkészlet, szöveges üzemmód stb.).

### 2.1.2. A botkormány-csatlakozó felhasználása mérésekhez

A számítógépbe nem csak a billentyűzeten keresztül lehet információt juttatni, hanem számos más bemeneti egységen keresztül is. Például: magnetofon, mágneslemez-egység (floppy), fényceruza, botkormány stb. Ezek közül legegyszerűbb a mindössze 5 kapcsolót tartalmazó, a játékokhoz szinte nélkülözhetetlen botkormány (joystick). A Commodore 16-hoz két botkormány is kapcsolható, és ezek állapota (azaz, hogy az öt kapcsoló közül melyek vannak zárva) a **JOY (1)** és a **JOY (2)** függvényekkel egyszerűen lekérdezhető.

Ha a csatlakozás kérdését valamilyen házilag készült csatlakozóval, vagy egy gyári botkormányról leszerelt csatlakozóval sikerül megoldani, a számítógép sokféle fizikai mérés közvetlen elvégzését teszi lehetővé. Megfelelő programmal például mérhetjük, hogy egy lejtőn elhelyezett érintkezősorozat egyes elemeit milyen időpontokban zárja a lejtőn mozgó fémgolyó. A

számítógéphez kapcsolt, és különböző hőmérsékleten kapcsoló bimetalok segítségével adott hőmérsékletek elérését érzékelheti a számítógép. (Ilyen bimetál a hőfokszabályozós vasalókból szerelhető ki.)

További lehetőség, hogy egy megfelelő fotoellenállást (esetleg egy tranzisztoron keresztül) kötünk a botkormány-csatlakozóra. Ebben az esetben a gép fény érzékelésére is alkalmassá válik. Egy ilyen fényérzékelőből és egy fényforrásból felépített fotokapuvval számos mozgástani mérést végezhetünk. Itt általában azt az időtartamot kell a programnak megmérnie, amíg a test eltakarja a fény útját. A test méretének ismeretében így a sebesség meghatározható.

A könyvben található SZABADESÉS MÉRÉS program szintén a botkormány-csatlakozón keresztül érzékeli a golyó indítását és a földre (kapcsolóra) esését. A kísérleti eszköz leírását és a megfelelő kapcsolási rajzot a program részletes ismertetőjénél megadjuk. Itt csak a botkormány-csatlakozók bekötését közöljük, hogy azt más kísérletek tervezésénél is fel lehessen használni. (2. ábra) A gépen lévő csatlakozót a rajz kívülről nézve ábrázolja. A mérések során az érzékelőelemek, kapcsolók egyik pólusát a 8. számú kivezetéshez, a másikat az 1-4. vagy a 6. számú kivezetések egyikére kell kötni. **A helyes bekötésre nagyon vigyázzunk**, mert az 5. számú kivezetésen az 5V-os tápfeszültség is megtalálható, és ezt más kivezetésekkel összekötve rövidzárlatot okozhatunk. A mérések során ügyeljünk arra is, hogy "idegen" feszültség ne jusson a csatlakozókra, mert ez is a gép tönkremeneteléhez vezethet.

## 2.2. Kezelési útmutatók és programlisták

A programokat a fizika fejezetei szerint csoportosítva közöljük. Minden program elején egy táblázat segítségével szeretnénk segítséget adni ahhoz, hogy a program milyen szervezési keretek között (osztály, csoport, egyéni) milyen didaktikai funkció megvalósítására alkalmazható. A táblázatban 0 jelzi, ha az adott szervezési keretek közt az adott funkció megvalósítható, és 0 szerepel ha erre kifejezetten ajánlott.

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 210 s (41 mp.)

Speciális karakter: @ → ■    £ → ♀

A program a  $Q = m / v$  összefüggés feladatokban történő alkalmazását gyakoroltatja. Elindítása után egy rövid ismertető jelenik meg. Ennek elolvasása után a T lenyomásával léphetünk tovább. A megjelenő menü alapján választhatjuk ki, hogy melyik mennyiség kiszámítását akarjuk gyakorolni. A választáshoz csak a megfelelő számot kell lenyomni. Itt nyílik lehetőség arra is, hogy a 9 gomb lenyomásával az addig elért eredményeket kiírassuk a képernyőre. A HELP megnyomásával a program elején található bevezetőhöz lehet visszalépni.

A megfelelő szám kiválasztása után egy feladat jelenik meg, melynek szövege, illetve adatai véletlenszerűen változnak. A HELP megnyomására a gép megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az elért eredménybe.

A választ a billentyűzet segítségével lehet beírni, és a RETURN billentyű megnyomásával kell lezárni. A válaszba a mértékegységet is be kell írni. A mértékegységeket rövidítve (g/cm<sup>3</sup>, g, cm<sup>3</sup>) vagy teljes nevükkel (gramm/köbcentiméter, gramm, köbcentiméter) is beírhatjuk. A hatványkitevő (3) a @ gombbal írható be.

A gép a választ elemzi, és hiba esetén annak jellegére utaló üzenet kiírása után megismétli a kérdést. Ha a második válasz is helytelen, a megoldást is kiírja a képernyőre. Helyes válasz esetén csak a HELYES felirat jelenik meg. Ezután az utolsó sorban megjelenő kiírásnak megfelelően az U gombbal új feladat kérhető. Az M gombbal a menühez térhetünk vissza, és más típusú feladatot választhatunk. A HELP billentyűvel ez esetben is a bevezetőhöz lehet visszamenni.

Az elért eredmények kiírását a menü alapján a 9 gomb lenyomásával kérhetjük. A képernyőn típusonként és összesítve is megjelenik a kapott feladatok száma, az elért pontszám és az eredmény (százalékban kifejezve). Egy feladatra 2, illetve 1 pont kapható aszerint, hogy az első vagy csak a második válasz volt a helyes.

A T gombbal a menühez, a HELP billentyű segítségével pedig a bevezetőhöz lehet visszajutni.

A program a RUN/STOP billentyűvel állítható le. A karakterkészlet módosítása miatt a képernyő a leállítás után elkövetett esetleges hiba következtében olvashatatlanává válik. Ezen az F1 billentyű megnyomásával segíthetünk. Ezzel az eljárással a program is a memóriában marad. A gép a program leállítás után is a módosított karakterkészletet használja, és a funkcióbillentyűk is átprogramozott állapotban maradnak. Ez a helyzet a RESET gomb megnyomásával szüntethető meg, de ez a megoldás a programot is törli a memóriából.

```
10 rem suruseg (gyak 1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 dimk$(9),t(9),p1(9),p2(9),d1(9),d2(9)
70 graphic0,1:restore
80 fori=0to21:reada:poke3048+i,a:next
90 fori=0to18:reada:poke4072+i,a:next
100 sys3048
110 fori=0to7:reada:poke15360+i,a:next
120 fori=0to7:reada:poke15584+i,a:next
130 poke65298,56:poke65299,56
140 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
150 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
160 data 112,8,48,8,112,0,0,0
170 data 60,102,102,124,48,24,12,56
180 key1,"poke65298,56:poke65299,56"+chr$(13)
190 fori=2to7:keyi,"":next:key8,chr$(140)
200 col=10:gosubl090
210 printchr$(14);chr$(8)
220 print"  [ ] [ ] [ ] [ ]"
230 print"  [ ] [ ] [ ] [ ] Suruseg (gyak 1)"
240 print"  [ ] [ ] [ ] [ ]"
250 print"  [ ]A program a  $f = m / V$  összefugges fel-"
260 print"adatokban torteno alkalmazasat gyakorol-tatja."
270 print"  [ ]A valaszokba a merkegegysegeket is be kell irni."
280 print"  [ ]A valaszt a RETURN gomb megnyomasaval kell lezarni."
290 print"  [ ]A program jegyzi az elert eredmenyt."
300 fori=1to9
310 readk$(i),t(i),p1(i),p2(i),d1(i),d2(i)
320 next
330 char1,25,24,"TOVABB = [ ]T[ ]"
340 getkeya$:ifa$="t"ora$="T"then350:elseifa$=chr$(140)then70:else340
350 scncrlr
360 col=7:gosubl090
370 char1,5,3,"1. Tomeg [ ]"
380 char1,20,4," [ ]"
```

```

390 char1,5,5,"2. Terfogat      > kiszamitasa"
400 char1,20,6,"|"
410 char1,5,7,"3. Suruseg      |"
420 char1,5,9,"4. Vegyes feladatok"
430 char1,5,13,"5. Tomeg       |"
440 char1,20,14,"| kiszamitasa"
450 char1,5,15,"6. Terfogat    >"
460 char1,20,16,"| szoveg alapjan"
470 char1,5,17,"7. Suruseg     |"
480 char1,5,19,"8. Vegyes szoveges feladatok"
490 char1,5,23,"9. EREDMENYEK"
500 getkeyt$:ift$=chr$(140)then70
510 t=val(t$)
520 ift<1ort>9then500
530 ift=9then1750
540 ift=1theni=1+3*rnd(0)
550 ift=2theni=4+3*rnd(0)
560 ift=3theni=7+3*rnd(0)
570 ift=4thent=int(1+3*rnd(0)):goto540
580 ift=5theni=1+3*rnd(0):t=4
590 ift=6theni=4+3*rnd(0):t=5
600 ift=7theni=7+3*rnd(0):t=6
610 ift=8thent=int(5+3*rnd(0)):goto580
620 :
630 k$=k$(i):p1=p1(i):p2=p2(i):d1=d1(i):d2=d2(i)
640 scnclr:col=11:gosub1090
650 x=int(34+66*rnd(0))+10↑(d1-2)
660 y=int(2+8*rnd(0))+10↑(d2-1)
670 z=x*y
680 x$=str$(x):ifx<1thenx$=" 0"+right$(x$,len(x$)-1)
690 y$=str$(y):ify<1theny$=" 0"+right$(y$,len(y$)-1)
700 z$=str$(z):ifz<1thenz$=" 0"+right$(z$,len(z$)-1)
710 r=0
720 ontgosub1130,1180,1230,1280,1340,1400
730 char1,7,7,"←"
740 a$="":gosub1470
750 ifa$=chr$(140)thenprint" ":r(t)=r(t)-r:goto950
760 h=0
770 ifabs(val(b$)-e)>0.0001thenh=1
780 n=1000000
790 ifabs(val(b$)-e*n)<0.0001andn<>1thenh=2
800 n=n/10:ifn>.0000001then790
810 lb=len(b$):l1=len(m1$):l2=len(m2$)
820 ifright$(b$,l1)<>m1$then840
830 iflb>l1+1thenifval(left$(b$,lb-l1-1))=val(b$)then860:else870
840 ifright$(b$,l2)<>m2$then860
850 iflb>l2+1thenifval(left$(b$,lb-l2-1))=val(b$)then860:else870
860 h=h+4
870 char1,0,10,""
880 if(hand1)=1thenprint"          Szamitasi hiba"
890 if(hand2)=2thenprint"          Nagysagrendi hiba"
900 if(hand4)=4thenprint"          Mertekegyseg hiba"
910 ifh=0then1040

```

```

920 r=r+1:r(t)=r(t)+1
930 tt=2000:gosub1560
940 ifr=1thenscnclr:goto720
950 char1,0,10,chr$(27)+"q"
960 char1,0,12,chr$(27)+"q"
970 ontgosub1580,1640,1700,1580,1640,1700
980 char1,7,24,"UJ FELADAT =  U  MENU =  M"
990 getkeya$:ifa$=chr$(140)then70
1000 ifa$="u"ora$="U"then510
1010 ifa$="m"ora$="M"then350
1020 goto990
1030 :
1040 j(t)=j(t)+2-r
1050 char1,7,14,"Helyes."
1060 tt=1000:gosub1560
1070 goto980
1080 :
1090 :
1100 color0,col,1:color1,col,7:color4,col,1
1110 return
1120 :
1130 char1,1,1,"£ =" +x$+" g/cm@"
1140 char1,1,3,"V =" +y$+" cm@"
1150 char1,1,5," _____"
1160 char1,1,7,"m = ? " :gotol320
1170 return
1180 char1,1,1,"m =" +z$+" g"
1190 char1,1,3,"£ =" +x$+" g/cm@"
1200 char1,1,5," _____"
1210 char1,1,7,"V = ? " :gotol380
1220 return
1230 char1,1,1,"m =" +z$+" g"
1240 char1,1,3,"V =" +y$+" cm@"
1250 char1,1,5," _____"
1260 char1,1,7,"£ = ? " :gotol440
1270 return
1280 char1,0,1,k$
1290 char1,0,5,"Mekkora a tomege?"
1300 char1,p1,1,x$
1310 char1,p2,1,y$
1320 e=z:m1$="g":m2$="gramm"
1330 return
1340 char1,0,1,k$
1350 char1,0,5,"Mekkora a terfogata?"
1360 char1,p1,1,z$
1370 char1,p2,1,x$
1380 e=y:m1$="cm":m2$="kobcentimeter"
1390 return
1400 char1,0,1,k$
1410 char1,0,5,"Mekkora a surusege?"
1420 char1,p1,1,z$
1430 char1,p2,1,y$
1440 e=x:m1$="g/cm":m2$="gramm/kobcentimeter"

```

```

1440 e=x:m1$="g/cm@":m2$="gramm/kobcentimeter"
1450 return
1460 :
1470 b$=""
1480 getkeya$:ifa$=chr$(140)thenreturn
1490 ifa$=chr$(13)andb$<>" "thenprint " ":return
1500 ifa$=chr$(20)andb$>" "thenprinta$;"← █";:b$=left$(b$,len(b$)
-1):gotol480
1510 if(a$<="9"anda$>="0")or(a$<="z"anda$>="@")or(a$<="Z"anda$>="
✓")then1540
1520 ifa$="."ora$="*"ora$="/"then1540
1530 gotol480
1540 printa$;"← █";:b$=b$+a$:iflen(b$)>25thenprint " ":return
1550 gotol480
1560 :
1570 fortt=ttto0step-1:next:return
1580 :
1590 char1,7,12,"m = £ * V"
1600 char1,7,14,"m =" +x$+" g/cm@ *"+y$+" cm@"
1610 char1,7,16,"m =" +z$+" g/cm@ * cm@"
1620 char1,7,18,"m =" +z$+" g"
1630 return
1640 :
1650 char1,7,12,"V = m / £"
1660 char1,7,14,"V =" +z$+" g /"+x$+" g/cm@"
1670 char1,7,16,"V =" +y$+" g / g/cm@"
1680 char1,7,18,"V =" +y$+" cm@"
1690 return
1700 :
1710 char1,7,12,"£ = m / V"
1720 char1,7,14,"£ =" +z$+" g /"+y$+" cm@"
1730 char1,7,16,"£ =" +x$+" g/cm@"
1740 return
1750 :
1760 scnc1r
1770 col=2:gosub1090
1780 char1,17,0,"feladat pont eredmeny"
1790 char1,0,3,"Tomeg"
1800 char1,0,5,"Terfogat"
1810 char1,0,7,"Suruseg"
1820 char1,0,11,"Tomeg (szov.)"
1830 char1,0,13,"Terfogat (szov.)"
1840 char1,0,15,"Suruseg (szov.)"
1850 char1,0,19,"OSSZESEN"
1860 fori=1to3
1870 j=j(i):n=j+r(i)
1880 char1,19,1+2*i,"":printusing"###";n/2
1890 char1,26,1+2*i,"":printusing"###";j
1900 ifn=0then1930
1910 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1920 char1,34,1+2*i,"":printusing"### %";j*100/n
1930 color1,2:next
1940 fori=4to6
1950 j=j(i):n=j+r(i)

```



```

1960 char1,19,3+2*i,"":printing"###";n/2
1970 char1,26,3+2*i,"":printing"###";j
1980 ifn=0then2010
1990 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
2000 char1,34,3+2*i,"":printing"### %";j*100/n
2010 color1,2:next
2020 color1,8,6
2030 j=0:r=0:fori=1to6:j=j+j(i):r=r+r(i):next:n=r+j
2040 char1,19,19,"":printing"###";n/2
2050 char1,26,19,"":printing"###";j
2060 ifn=0then2080
2070 char1,34,19,"":printing"### %";j*100/n
2080 color1,2
2090 goto330
2100 data"Egy cm@ terfogatu femgolyo g/cm@ surusegu anyag
bol keszult."
2110 data1,29,3,1,2
2120 data"Egy test g/cm@ s rusegu, cm@ terfogatu."
2130 data1,8,28,1,2
2140 data"Egy ko terfogata cm@ es g/cm@ a surusege."
2150 data1,26,16,1,2
2160 data"Egy ontveny tomege g es g/cm@ surusegu anyag
bol keszult."
2170 data2,18,27,1,2
2180 data"Egy fahasab g tomege es g/cm@ a surusege."
2190 data2,11,27,0,3
2200 data"Egy uvegben g tomege es g/cm@ surusegu folya
dek van."
2210 data2,11,27,0,3
2220 data"Egy csomag tomege g es cm@ a terfogata."
2230 data3,17,26,0,3
2240 data"Egy cm@-es gepalkatresznek g a tomege."
2250 data3,30,3,1,3
2260 data"Egy femgolyo terfogata cm@ es g a tomege."
2270 data3,32,22,1,2

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 200 s (40 mp.)

Speciális karakter: @ → ⓐ    £ → Ⓔ

A program a sűrűség fogalmának elmélyítéséhez készült. A  $\rho = m / v$  összefüggésben szereplő három mennyiség közül a gép 2 - 2 mennyiség nagyságrendi viszonyait megadja, és ennek ismeretében kell a tanulóknak a hiányzó mennyiségeket összehasonlítani. (pl. a gép megadja, hogy a két test tömege egyenlő és az egyik térfogata 3-szor nagyobb a másik térfogatánál. Ebben az esetben a sűrűségeket kell összehasonlítani.)

Az elindítás után rövid tájékoztatót ad a program. Ezt elolvasva a T gomb lenyomásával lehet továbblépni. Ezután történik a nehézségi fokozat kiválasztása. A felsorolt lehetőségek közül a megfelelő szám lenyomásával lehet választani. Itt van mód arra, hogy az addig elért eredményt a 4 gomb lenyomásával kiírassuk a képernyőre. A HELP billentyű lenyomásával a bevezetőhöz lehet visszamenni.

A nehézségi fokozat meghatározása után az összehasonlítandó mennyiségeket kell kiválasztani. Itt is a megfelelő szám lenyomásával lehet választani. Ezután a program a választásokat figyelembevéve véletlenszerű értéket ad 2 - 2 mennyiségnek, és rákérdez a hiányzó mennyiségek összehasonlításának eredményére. A választ a < = > és a ? gombok segítségével lehet beírni. A ? gombot akkor kell lenyomni, ha a kérdésre nem adható egyértelmű válasz.

A HELP billentyű megnyomására a program megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az eredménybe. A válasz beírása után, ha az hibás volt, a program az erre utaló üzenet kiírása után megadja a helyes választ, ellenkező esetben csak a REMEK felirat jelenik meg a képernyőn.

Ezután az U gomb lenyomásával új feladatot kérhetünk, az M gomb lenyomásával pedig az első menühöz (nehézségi fokozat kiválasztása és eredmények) mehetünk vissza. A HELP billentyű megnyomása ebben az esetben a bevezetőhöz való visszatérést eredményezi.

Az eredmények kiírását az említett módon az első menü alapján a 4 gomb lenyomásával kérhetjük. A program a nehézségi fokozat és az összehasonlítandó mennyiség szerint csoportosítva, valamint összesítve százalékban adja meg az elért eredményt.

Az eredmények kiírása után a T gombbal újra a menühöz lehet visszajutni, míg a HELP billentyű lenyomásával a bevezető ismertetéshez léphetünk vissza.

A program a RUN/STOP billentyűvel állítható le. A karakterkészlet átírása miatt a leállítás után elkövetett esetleges hiba miatt a képernyő olvashatatlanul válhat. Ezen az F1 gomb lenyomásával segíthetünk, és ezzel a program sem törlődik. A gép dolgozik és a program megállítása után a funkcióbillentyűk is átállított állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez a program törlését eredményezi.

```
10 rem suruseg (gyak 2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 fori=0to7:reada:poke15584+i,a:next
120 poke65298,56:poke65299,56
130 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
140 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
150 data112,8,48,8,112,0,0,0
160 data60,102,102,124,48,24,12,56
170 key1,"poke65298,56:poke65299,56"+chr$(13)
180 fori=2to7:keyi,"":next:key8,chr$(140)
190 fok$(1)="Konnyu ":t$(1)="Tomeg"
200 fok$(2)="Kozepes":t$(2)="Terfogat"
210 fok$(3)="Nehez ":t$(3)="Suruseg"
220 t$(4)="VEGYES"
230 col=11:gosub2160
240 printchr$(14);chr$(8)
250 print"  [ ] [ ] [ ] [ ]"
260 print"  [ ] [ ] [ ] [ ] Suruseg (gyak.2)"
270 print"  [ ] [ ] [ ] [ ]"
280 print"AAAA program az m = k * V osszefugges"
290 print"elemzesehez keszult."
300 print"AA megadott adatok alapjan ossze kell"
310 print"hasonlítani a ket mennyiseget."
320 print"AAz összehasonlítás eredményét a < = >"
330 print"gombokkal lehet beírni."
340 print"AAHa nem adható egyértelmű válasz, akkor"
350 print"a ? gombot kell megnyomni."
360 char1,25,24,"TOVABB = [ ]"
```

```

370 getkeya$:ifa$=chr$(140)thenrun
380 ifa$<>"t"anda$<>"T"then370
390 scnclr
400 col=10:gosub2160
410 char1,5,0,"NEHEZSEGI FOKOZAT VALASZTAS"
420 char1,14,7,"1. "+fok$(1)
430 char1,14,10,"2. "+fok$(2)
440 char1,14,13,"3. "+fok$(3)
450 char1,14,18,"4. EREDMENYEK"
460 char1,2,24,"A szam lenyomasaval lehet valasztani."
470 getkeya$:ifa$=chr$(140)thenrun
480 ifa$="4"then1880
490 ifa$<"1"ora$>"3"then470
500 f=val(a$)
510 scnclr
520 col=10:gosub2160
530 gosub2190
540 char1,4,7,"AZ OSSZEHASONLITANDO MENNYISEG"
550 char1,16,12,"1. "+t$(1)
560 char1,16,14,"2. "+t$(2)
570 char1,16,16,"3. "+t$(3)
580 char1,16,20,"4. "+t$(4)
590 char1,2,24,"A szam lenyomasaval lehet valasztani."
600 getkeyt$:ift$=chr$(140)thenrun
610 ift$<"1"ort$>"4"then600
620 t=val(t$):ift=4thent=1+3*rnd(0)
630 scnclr
640 col=11:gosub2160
650 gosub2190
660 x1=int(1+9*rnd(0)):x2=int(1+9*rnd(0))
670 y1=int(1+9*rnd(0)):y2=int(1+9*rnd(0))
680 r=rnd(0)
690 ifr<.3then720
700 ifr<.65thenx1=x2
710 ifr>=.65theny1=y2
720 ontgosub950,1260,1570
730 char1,8,5,x1$:char1,27,5,x2$
740 char1,8,9,y1$:char1,27,9,y2$
750 char1,7,11,"-----"
760 char1,8,13,z1$:char1,27,13,z2$
770 char1,19,13,"  ?  "
780 getkeya$:ifa$=chr$(140)then880
790 ifa$=","ora$="."ora$="/"thena$=chr$(asc(a$)+16)
800 ifa$<>"<"anda$<>"="anda$<>">"anda$<>"?"then780:elsechar1,19,1
3,a$
810 n(f,t)=n(f,t)+1
820 ifa$=e$then830:else860
830 j(f,t)=j(f,t)+1
840 char1,17,18,"REMEK"
850 goto900
860 char1,17,16,"ROSSZ"
870 fort=0to300:next
880 char1,11,20,"A helyes valasz:  "e$+"
890 fort=0to900:next

```

```

900 char1,5,24,"UJ FELADAT = U           MENU = M"
910 getkeya$:ifa$=chr$(140)thenrun
920 ifa$="m"ora$="M"then390
930 ifa$="u"ora$="U"then620
940 goto910
950 :
960 z1$=" m▯1":z2$=" m▯2"
970 onfgoto980,1030,1140
980 :
990 x1$=str$(x1)+"g/cm@":x2$=str$(x2)+"g/cm@"
1000 y1$=str$(y1)+"cm@":y2$=str$(y2)+"cm@"
1010 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
1020 return
1030 :
1040 x1$=str$(x1)+"£":x2$=str$(x2)+"£"
1050 ifx1=1thenx1$=" £"
1060 ifx2=1thenx2$=" £"
1070 ifx1=x2thenx1$=" £":x2$=x1$
1080 y1$=str$(y1)+"V":y2$=str$(y2)+"V"
1090 ify1=1theny1$=" V"
1100 ify2=1theny2$=" V"
1110 ify1=y2theny1$=" V":y2$=y1$
1120 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
1130 return
1140 :
1150 x1$=" £▯1":x2$=" £▯2"
1160 y1$=" V▯1":y2$=" V▯2"
1170 ifx1=x2andy1=y2thene$="=":gotol210
1180 ifx1<=x2andy1<=y2thene$="<":gotol210
1190 ifx1>=x2andy1>=y2thene$=">":gotol210
1200 e$="?"
1210 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1220 char1,19,5,r$
1230 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1240 char1,19,9,r$
1250 return
1260 :
1270 z1$=" V▯1":z2$=" V▯2"
1280 onfgotol300,1340,1450
1290 :
1300 x1$=str$(x1)+"g":x2$=str$(x2)+"g"
1310 y1$=str$(y1)+"g/cm@":y2$=str$(y2)+"g/cm@"
1320 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$=">"
1330 return
1340 :
1350 x1$=str$(x1)+"m":x2$=str$(x2)+"m"
1360 ifx1=1thenx1$=" m"
1370 ifx2=1thenx2$=" m"
1380 ifx1=x2thenx1$=" m":x2$=x1$
1390 y1$=str$(y1)+"£":y2$=str$(y2)+"£"

```

```

1400 ify1=1theny1$=" £"
1410 ify2=1theny2$=" £"
1420 ify1=y2theny1$=" £":y2$=y1$
1430 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1440 return
1450 :
1460 x1$=" m1":x2$=" m2"
1470 y1$=" £1":y2$=" £2"
1480 ifx1=x2andy1=y2thene$="=":goto1520
1490 ifx1<=x2andy1>=y2thene$="<":goto1520
1500 ifx1>=x2andy1<=y2thene$=">":goto1520
1510 e$="?"
1520 ifx1<y2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1530 char1,19,5,r$
1540 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1550 char1,19,9,r$
1560 return
1570 :
1580 z1$=" £1":z2$=" £2"
1590 onfgoto1600,1650,1760
1600 :
1610 x1$=str$(x1)+"g":x2$=str$(x2)+"g"
1620 y1$=str$(y1)+"cm@":y2$=str$(y2)+"cm@"
1630 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1640 return
1650 :
1660 x1$=str$(x1)+"m":x2$=str$(x2)+"m"
1670 ifx1=1thenx1$=" m"
1680 ifx2=1thenx2$=" m"
1690 ifx1=x2thenx1$=" m":x2$=x1$
1700 y1$=str$(y1)+"V":y2$=str$(y2)+"V"
1710 ify1=1theny1$=" V"
1720 ify2=1theny2$=" V"
1730 ify1=y2theny1$=" V":y2$=y1$
1740 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1750 return
1760 :
1770 x1$=" m1":x2$=" m2"
1780 y1$=" V1":y2$=" V2"
1790 ifx1=x2andy1=y2thene$="=":goto1830
1800 ifx1<=x2andy1>=y2thene$="<":goto1830
1810 ifx1>=x2andy1<=y2thene$=">":goto1830
1820 e$="?"
1830 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1840 char1,19,5,r$
1850 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1860 char1,19,9,r$
1870 return
1880 :
1890 col=15:go .b2160

```

```

1900 scnc1r
1910 char1,15,0,"EREDMENYEK"
1920 char1,14,5,"1.      2.      3.      OSSZ."
1930 char1,0,9,t$(1)
1940 char1,0,12,t$(2)
1950 char1,0,15,t$(3)
1960 char1,0,19,"OSSZESEN"
1970 forf=1to3
1980 n(4,t)=0:j(4,t)=0
1990 forf=1to3
2000 ifn(f,t)>0thene(f,t)=100*j(f,t)/n(f,t)
2010 n(4,t)=n(4,t)+n(f,t)
2020 j(4,t)=j(4,t)+j(f,t)
2030 next
2040 ifn(4,t)>0thene(4,t)=100*j(4,t)/n(4,t)
2050 next
2060 forf=1to4
2070 n(f,4)=0:j(f,4)=0
2080 forf=1to3
2090 ifn(f,t)>0thenchar1,5+7*f,6+3*t,"":printusing"###%";e(f,t)
2100 n(f,4)=n(f,4)+n(f,t)
2110 j(f,4)=j(f,4)+j(f,t)
2120 next
2130 ifn(f,4)>0thenchar1,5+7*f,19,"":printusing"###%";100*j(f,4)/
n(f,4)
2140 next
2150 goto360
2160 :
2170 color0,col,1:color1,col,7:color4,col,1
2180 return
2190 :
2200 char1,0,0," "
2210 char1,0,1," Fokozat: "+fok$(f)+" "
2220 char1,0,2," "
2230 return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 105 s (31 mp.)

Speciális karakter: @-e

A program a sűrűség fogalmának elmélyítéséhez használható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10 féle anyagra (alkohol, alumínium, arany, gyémánt, higany, jég, lítium, ólom, vas, víz) vonatkozó adatok szerepelnek. Az első oszlopban az anyag neve, a másodikban az adott térfogathoz tartozó tömeg, a harmadikban a térfogat (kezdeti értéke 10 cm<sup>3</sup>), az utolsóban pedig a sűrűség értéke található. Az alkohol tömegére vonatkozó adat előtt álló > jel a kurzor-mozgató billentyűkkel (nyilakkal) bármelyik anyag tömege vagy térfogata elé vihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN billentyű lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban (sötét alapon világos betűk) jelenik meg a képernyőn.

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Például a ↓ gomb kétszeri és a → gomb egyszeri megnyomása után az arany térfogata módosítható. Ha ekkor 100-at írunk be, és megnyomjuk a RETURN billentyűt, a 3. sorban az arany tömegénél 1932.00, térfogatnál 100.0 és a sűrűségnél 19.32 fog megjelenni. Ezzel az eljárással egy kiválasztott anyag esetén a tömeg és térfogat közti egyenes arányosság egyszerűen tanulmányozható. Lehetőség van valamennyi anyag tömegének, illetve térfogatának egyszerre történő megváltoztatására is. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja. Ezzel lehetőség nyílik azonos térfogatú (illetve azonos tömegű) anyagok tömegének (illetve térfogatának) az összehasonlítására.



A programban szereplő anyagok a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni a kívánt anyag nevét, majd az idézőjelen kívülre, a névtől egy vesszővel elválasztva a sűrűséget. Figyelem, a tizedesvessző helyett tizedespontot kell írni! Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A karakterkészlet átírása miatt a képernyő a leállítás után elkövetett esetleges hiba miatt olvashatatlanul válhat. Ez a helyzet az F1 billentyű megnyomásával megszüntethető. Ezzel a módszerrel a program sem törlődik. A gép a program leállítása után is a módosított karakterkészletet használja, továbbá a funkció-billentyűk is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ezzel a program is törlődik.

```
10 rem suruseg (tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 poke65298,56:poke65299,56
120 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
130 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
140 data112,8,48,8,112,0,0,0
150 key1,"poke65298,56:poke65299,56"+chr$(13)
160 fori=2to7:keyi,"":next
170 key8,chr$(140)
180 color0,2,6:color1,1,0:color4,2,6
190 printchr$(14);chr$(8)
200 print"  ██  ████  ████  "
210 print"  ██  ████  ████  █  SURUSEG (tabl.1.)"
220 print"  ██  ████  ████  "
230 print"███A programmal 10 kulonbozo anyag surusege"
240 print"tanulmanyozhato."
250 print"███A > jel a billentyuzeten talalhato"
260 print"nyilakkal a megfelelo helyre viheto."
270 print"███A kivant ertekek beirasa utan a RETURN"
280 print"gomb megnyomasara a program atirja a"
290 print"tablazat adatait."
300 print"███A beirt ertekek legfeljebb 4 karakterbol"
310 print"allhat."
320 print"███Az ESC billentyu megnyomasa utan az"
330 print"oszlop erteke atirhato."
340 char1,25,24,"TOVABB = ████"
350 fori=0to9:readnev$(i),z(i):next
360 getkeya$:ifa$=chr$(140)thenrun
```

```

370 ifa$<>"t"anda$<>"T"then360
380 color0,10,2:color1,2,7:color4,10,2
390 fori=0to9:y(i)=10:x(i)=y(i)*z(i):next
400 p=1:q=0
410 scnclr
420 print"          TOMEQ      TERFOGAT      SURUSEG"
430 print"          (g)          (cm@)          (g/cm@)"
440 fori=0to9:gosub620:next
450 b$=""
460 char1,10*p,4+2*q,">"
470 getkeya$:ifa$=chr$(140)thenrun
480 ifa$=" "andp=2andb$=""thenp=1:print" ":goto450
490 ifa$=" "andp=1andb$=""thenp=2:print" ":goto450
500 ifa$=" "andq>0andb$=""thenq=q-1:print" ":goto450
510 ifa$=" "andq<9andb$=""thenq=q+1:print" ":goto450
520 ifa$=" "andb$=""thenprint" ":p=1:q=0:goto450
530 ifa$=" "andb$=""thenp=2:fori=0to9:y(i)=10:gosub620:next:p=1:g
oto450
540 ifa$=chr$(27)andb$=""thengosub710:goto450
550 if(a$<"0"ora$>"9")anda$<>". "then470
560 i=q
570 print"          "
580 char1,10*p,4+2*i,"> "
590 gosub800
600 gosub870
610 goto450
620 :
630 ifp=1theny(i)=x(i)/z(i)
640 ifp=2thenx(i)=y(i)*z(i)
650 char1,0,4+2*i,nev$(i)
660 char1,10,4+2*i,""
670 printusing"#####. # ";x(i);
680 printusing"#####. # ";y(i);
690 printusing"#####.## ";z(i);
700 return
710 :
720 print" " :char1,10*p,24,"> "
730 gosub800
740 fori=0to9:gosub870:next
750 char1,10*p,24," "
760 q=0
770 return
780 :
790 getkeya$
800 ifa$=chr$(140)thenrun
810 ifa$=chr$(13)andb$>" "then860
820 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):print" ";;g
oto790
830 if(a$>"9"ora$<"0")anda$<>". "then790
840 b$=b$+a$
850 print" "+a$;" ";:iflen(b$)<4then790
860 return
870 :

```

```
880 b=val(b$)
890 ifp=1thenx(i)=b:elsey(i)=b
900 gosub620
910 return
920 :
930 data"Alkohol",0.79
940 data"Aluminium",2.7
950 data"Arany",19.32
960 data"Gyemant",3.5
970 data"Higany",13.59
980 data"Jeg 0 fok",0.92
990 data"Litium",0.53
1000 data"Olom",11.34
1010 data"Vas",7.87
1020 data"Viz 4 fok",1
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 200 s (40 mp.)

Speciális karakter: @ → <sup>3</sup> £ → e

A program a  $\varrho = m / v$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább a menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, aminek az értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl: kezdőértéknek 100-at, lépésközt 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységeket nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezekután három lehetőség közül lehet választani. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN billentyű megnyomásával az előzőleg használt értéket megtartsuk. A program futása a RUN/STOP billentyű megnyomásával szakítható meg. Mivel a gép a program leállítása után is a módosított karakterkészletet használja, egy esetleges hiba a képernyő olvashatatlaná válását eredményezi. Ez az állapot az F1 gomb megnyomásával megszüntethető. A funkcióbillentyűk átprogramozott állapota és a karakterkészlet a RESET gomb megnyomásával állítható vissza az eredeti állapotba, de ez a program törlését eredményezi.

```

10 rem suruseg (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next:rem @
110 fori=0to7:reada:poke15584+i,a:next:rem £
120 poke65298,56:poke65299,56
130 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
140 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
150 data112,8,48,8,112,0,0,0
160 data60,102,102,124,48,24,12,56
170 key1,"poke65298,56:poke65299,56"+chr$(13)
180 fori=2to7:keyi,"":next:key8,chr$(140)
190 f1$="          TOMEG          TERFOGAT          SURUSEG"
200 f2$="          (g)          (cm@)          (g/cm@)"
210 v1$="TOMEG"
220 v2$="TERFOGAT"
230 v3$="SURUSEG"
240 v$="-----"
250 sp$=" "
260 col=10:gosub2010
270 printchr$(14);chr$(8)
280 print"  "
290 print"  " SURUSEG (tabl.2.)"
300 print"  "
310 print"  A program a £ = m / V osszefugges elem-"
320 print"zesehez keszult."
330 print"  A három mennyiség közül egyet allando"
340 print"ertekunek valasztva egy másik mennyiség"
350 print"erteke szabadon változtathato."
360 print"  A program ezek figyelembevetelevel"
370 print"kiszamítja a harmadik változo megfelelo"
380 print"ertekeit."
390 print"  A CLEAR/HOME megnyomasara uj ertekek"
400 print"adhatok az egyes mennyisegeknek, a"
410 print"SHIFT es CLEAR/HOME egyutt es lenyomasa"

```

```

420 print"utan pedig az egyes mennyisegek szerepe"
430 print"valtoztathato meg."
440 char1,25,24,"TOVABB = T"
450 getkeya$:ifa$=chr$(140)thenrun
460 ifa$<"t"anda$<"T"then450
470 col=10:gosub2010
480 all=50:kezd=10:lep=1
490 print" " +sp$ "          A VIZSGALT ESET KIVALASZTASA          "+sp$
500 char1,7,5,"FUGGO          FUGGETLEN          "
510 char1,18,6,"ALLANDO"
520 char1,6,7,"VALTOZO          VALTOZO          "
530 char1,0,8,v$
540 char1,0,9," 1.   TOMEG          TERFOGAT          SURUSEG          "
550 char1,0,11," 2.   TOMEG          SURUSEG          TERFOGAT          "+v$
560 char1,0,13," 3.   TERFOGAT          TOMEG          SURUSEG          "
570 char1,0,15," 4.   TERFOGAT          SURUSEG          TOMEG          "+v$
580 char1,0,17," 5.   SURUSEG          TOMEG          TERFOGAT          "
590 char1,0,19," 6.   SURUSEG          TERFOGAT          TOMEG          "+v$
600 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET          VALASZTA
NI."
610 getkeyt$:ift$=chr$(140)thenrun
620 ift$<"1"ort$>"6"then610
630 col=15:gosub2010
640 print" " ;sp$ ;"          ERTEKADAS          " ;sp$
650 char1,2,5,"FUGGO VALTOZO          :"
660 char1,2,7,"Allando mennyiseg          :"
670 char1,2,9,"Fuggetlen valtozo          :"
680 onval(t$)goto690,870,1050,1230,1410,1590
690 :
700 char1,22,5,v1$
710 char1,22,7,v2$
720 char1,22,9,v3$
730 char1,0,11,v$
740 char1,0,14,"A terfogat erteke (cm@)          :          "
750 a=all:p=26:q=14:gosub1770:all=a
760 char1,0,17,"A suruseg"
770 char1,5,19,"- kezderteke (g/cm@)          :          "
780 a=kezd:p=29:q=19:gosub1770:kezd=a
790 char1,5,21,"- lepeskoz (g/cm@)          :          "
800 a=lep:p=26:q=21:gosub1770:lep=a
810 col=9:gosub2010
820 char1,5,1," "
830 forj=0to9
840 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
850 next
860 goto1900
870 :
880 char1,22,5,v1$
890 char1,22,7,v3$
900 char1,22,9,v2$
910 char1,0,11,v$
920 char1,0,14,"A suruseg erteke (g/cm@)          :          "
930 a=all:p=27:q=14:gosub1770:all=a

```

```

940 char1,0,17,"A terfogat"
950 char1,5,19,"- kezdoerteke (cm@) : "
960 a=kezd:p=27:q=19:gosub1770:kezd=a
970 char1,5,21,"- lepeskoz (cm@) : "
980 a=lep:p=24:q=21:gosub1770:lep=a
990 col=9:gosub2010
1000 char1,5,1,"  ────  "
1010 forj=0to9
1020 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
1030 next
1040 goto1900
1050 :
1060 char1,22,5,v2$
1070 char1,22,7,v1$
1080 char1,22,9,v3$
1090 char1,0,11,v$
1100 char1,0,14,"A tomeg erteke (g) : "
1110 a=all:p=21:q=14:gosub1770:all=a
1120 char1,0,17,"A suruseg"
1130 char1,5,19,"- kezdoerteke (g/cm@) : "
1140 a=kezd:p=29:q=19:gosub1770:kezd=a
1150 char1,5,21,"- lepeskoz (g/cm@) : "
1160 a=lep:p=26:q=21:gosub1770:lep=a
1170 col=9:gosub2010
1180 char1,15,1,"  ────  "
1190 forj=0to9
1200 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1210 next
1220 goto1900
1230 :
1240 char1,22,5,v2$
1250 char1,22,7,v3$
1260 char1,22,9,v1$
1270 char1,0,11,v$
1280 char1,0,14,"A suruseg erteke (g/cm@) : "
1290 a=all:p=27:q=14:gosub1770:all=a
1300 char1,0,17,"A tomeg"
1310 char1,5,19,"- kezdoerteke (g) : "
1320 a=kezd:p=25:q=19:gosub1770:kezd=a
1330 char1,5,21,"- lepeskoz (g) : "
1340 a=lep:p=22:q=21:gosub1770:lep=a
1350 col=9:gosub2010
1360 char1,15,1,"  ────  "
1370 forj=0to9
1380 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1390 next
1400 goto1900
1410 :
1420 char1,22,5,v3$
1430 char1,22,7,v1$
1440 char1,22,9,v2$
1450 char1,0,11,v$
1460 char1,0,14,"A tomeg erteke (g) : "

```

```

1470 a=all:p=21:q=14:gosub1770:all=a
1480 char1,0,17,"A terfogát"
1490 char1,5,19,"- kezderteke (cm@) : "
1500 a=kezd:p=27:q=19:gosub1770:kezd=a
1510 char1,5,21,"- lepeskoz (cm@) : "
1520 a=lep:p=24:q=21:gosub1770:lep=a
1530 col=9:gosub2010
1540 char1,28,1," ┌──────────┐"
1550 forj=0to9
1560 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1570 next
1580 goto1900
1590 :
1600 char1,22,5,v3$
1610 char1,22,7,v2$
1620 char1,22,9,v1$
1630 char1,0,11,v$
1640 char1,0,14,"A terfogát erteke (cm@) : "
1650 a=all:p=26:q=14:gosub1770:all=a
1660 char1,0,17,"A tomeg"
1670 char1,5,19,"- kezderteke (g) : "
1680 a=kezd:p=25:q=19:gosub1770:kezd=a
1690 char1,5,21,"- lepeskoz (g) : "
1700 a=lep:p=22:q=21:gosub1770:lep=a
1710 col=9:gosub2010
1720 char1,28,1," ┌──────────┐"
1730 forj=0to9
1740 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1750 next
1760 goto1900
1770 :
1780 b$="":char1,p,q," "
1790 print" ←┌───┐";getkeya$
1800 ifa$=chr$(13)then1840
1810 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$a$
1820 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1830 iflen(b$)<6then1790
1840 ifb$=""then1870
1850 ifval(b$)<.01thenchar1,p,q,sp$:goto1770
1860 a=val(b$)
1870 char1,p,q,str$(a)+sp$
1880 ifa<1thenchar1,p,q,"0"
1890 return
1900 :
1910 char1,0,0,f1$:char1,0,3,f2$
1920 forj=0to9
1930 char1,0,6+2*j," "
1940 printusing"#####.#";x(j);
1950 printusing"#####.#";y(j);
1960 printusing"#####.##";z(j);
1970 next

```



```
1980 getkeya$:ifa$="☐"then630:elseifa$=""then470:elseifa$=chr$(14
0)thenrun
1990 goto1980
2000 stop
2010 :
2020 scnclr
2030 color0,col,6:color1,col,1:color4,col,6
2040 return
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 85 s  
Speciális karakter: nincs

A program segítségével az egyenletes mozgás szimulálható, és használata hozzájárulhat a sebesség fogalmának jobb megértéséhez is. Az indítás után megjelenő rövid ismertetőt elolvasva választani kell, hogy a mozgás nyomképét megrajzolja-e a program. Ehhez csupán az I illetve N gombok egyikét kell megnyomni.

Ha ez megtörtént, a program felrajzolja az út - idő grafikonhoz szükséges koordinátarendszert, s a testet jelképező kis négyzetet, és megjelenik a START = S felirat. Az S gomb megnyomására a test véletlenszerűen választott sebességgel, egyenletes mozgást végezve halad. A program közben berajzolja az út- idő grafikon néhány pontját, és ha kértük, akkor az ezeknek megfelelő nyomképeket is. Miután a test elérte a képernyő tetejét, a megjelenő feliratoknak megfelelően több lehetőség közül választhatunk:

Az U gomb megnyomásával a mozgás újra megismételhető, míg a T gomb lenyomására a program továbblép egy más sebességgel történő mozgás vizsgálatához. Az új mozgás út - idő grafikonját a program ugyanabban a koordinátarendszerben ábrázolja, ahol az előzőt, így a különböző sebességű mozgások grafikonjai közvetlenül összehasonlíthatók. Ha az előzőleg megrajzolt grafikonokat törölni akarjuk, a SHIFT és a CLEAR/HOME billentyűket kell egyidejűleg lenyomni.

A G gomb megnyomásával "görbét" illeszthetünk az út - idő grafikon bejelölt pontjaihoz. A grafikon megrajzolásával egyidőben az út, az idő és a sebesség pillanatnyi értékét is kiírja a program. A grafikon kirajzolása után már csak az ÚJRA=U és a TOVÁBB=T felirat jelenik meg.

A HELP billentyű megnyomása itt is a bevezetőhöz történő visszatérést eredményezi.

A program a RUN/STOP billentyű megnyomásával állítható le. Ha a leállítás grafikus üzemmódban történik, és ezért a READY üzenet és a villogó kurzor nem látható, a GRAPHIC 0 parancs "vakon" történő

beírásával és a RETURN billentyű megnyomásával visszatérhetünk a szöveges üzemmódnhoz. A funkcióbillentyűk eredeti állapota a RESET gomb megnyomásával visszaállítható, de ez a program törlésével jár együtt.

```
10 rem egyenletes mozgás
20 rem zatonyi sandor bekescsaba 1986.
30 deffns(t)=int(180.5-v*t):fort=1to7:keyt,"":next:key8,chr$(8):p
oke740,212
40 sp$="          ":s$="START=S":t$="TOVABB=T":u$="UJRA=U":g$="GR
AFIKON=G"
50 color0,9,2:color4,9,2:color1,2,7
60 graphic0,1:printchr$(14);chr$(8);
70 print"EGYENLETES MOZGAS"
80 print"A gép szemlelteti az egyenletes mozgást"
90 print"és felrajzolja az ut - idő grafikonon néhány pontját."
100 print"Ezekhez a G gomb megnyomására grafikont rajzol."
110 print"A mozgás az U gomb megnyomásával újra"
120 print"indul, a T hatására pedig másik mozgás vizsgálható."
130 char1,13,24,"NYOMKEPET RAJZOLJAK? (I/N)"
140 geta$:ifa$="i"ora$="I"thenj=1:elseifa$="n"ora$="N"thenj=0:els
el40
150 graphic1,1:h=180:fi=270:locate320,180:gosub380
160 h=7.1:fi=45:locatel135,5:gosub380:locate314,175:gosub380
170 fort=20to180step20:draw1,137,tto140,t:draw1,120+t,180to120+t,
183:next
180 char1,18,1,"s(m)":char1,35,21,"t(s)"
190 char1,17,24,"0      5"
200 char1,15,22,"0":char1,14,10,"50"
210 v=int(11*rnd(0)+3)/5:g=0
220 t=0:s=180:n=1:gosub390:char1,0,0,s$:getkeya$:ifa$=chr$(8)then
run
230 ifa$<>"s"anda$<>"S"then220:elsechar1,0,0,sp$
240 dountils<0:n=j:gosub390:t=t+20:s=fns(t):n=1:gosub390:draw1,14
0+t,s:loop
250 ifg=0thenchar1,0,9,g$
260 char1,0,11,u$:char1,0,13,t$
270 getkeya$:ifa$=chr$(8)thenrun:elseif(a$="g"ora$="G")andg=0then
300
280 ifa$="u"then150:elseifa$="u"ora$="U"thengosub400:goto220
290 ifa$="t"ora$="T"thengosub400:goto210:else270
300 fort=9to24:char1,0,t,sp$:next:t=5:draw1,140,180
310 char1,0,20,"s =      m"
320 char1,0,22,"t =      s"
330 char1,0,24,"v =      m/s"
340 dountilt>180orfns(t)<0:drawto140+t,fns(t)
350 c=0:m=v*t/2:gosub410:c=2:m=t/20:gosub410:c=4:m=v*10:gosub410
360 char1,0,13,"TOVABB=G":geta$:ifa$<>"g"anda$<>"G"then360
370 t=t+5:loop:g=1:goto260
380 drawtoh;fitoh;fi+90:fi=fi+90:return
390 boxn,90,s,100,s-10:return
400 fort=0to24:char1,0,t,sp$+"          ":next:return
410 a$=str$(m)+sp$:ifm<1thena$=" 0"+mid$(a$,2,3)
420 char1,3,20+c,left$(a$,5):return
```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 195 s (39 mp.)

Speciális karakter: nincs

A program a  $v = s / t$  összefüggés feladatokban történő alkalmazását gyakoroltatja. A program elindítása után egy rövid ismertető jelenik meg. Ennek elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő menü alapján választhatjuk ki, hogy melyik mennyiség kiszámítását akarjuk gyakorolni. A választáshoz csak a megfelelő számot kell lenyomni. Itt nyílik lehetőség arra is, hogy a 9 gomb lenyomásával az addig elért eredményeket kiírassuk a képernyőre. A HELP billentyű megnyomásával a program elején található bevezetőhöz lehet visszalépni.

A megfelelő mennyiség kiválasztása után egy feladat jelenik meg, melynek szövege, illetve adatai véletlenszerűen változnak. A HELP billentyű megnyomására a gép megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az eredménybe.

A választ a billentyűzet segítségével lehet beírni, és a RETURN billentyű megnyomásával kell lezárni. A válaszba a mértékegységet is be kell írni. A mértékegységeket rövidítve (m/s, m, s) vagy teljes nevükkel (méter/másodperc, méter, másodperc) is beírhatjuk. A gép a választ elemzi, és hiba esetén annak jellegére utaló üzenet kiírása után megismétli a kérdést. Ha a második válasz is helytelen, a megoldást is kiírja a képernyőre. Helyes válasz esetén csak a HELYES felirat jelenik meg. Ezután az utolsó sorban megjelenő kiírásnak megfelelően az U gombbal új feladat kérhető, vagy az M gombbal a menühez térhetünk vissza, és más típusú feladatot választhatunk. A HELP billentyűvel ez esetben is a bevezetőhöz lehet visszamenni.

Az elért eredmények kiírását a menü alapján a 9 gomb lenyomásával kérhetjük. A képernyőn típusonként és összesítve is megjelenik a kapott feladatok száma, az elért pontszám és az eredmény százalékban kifejezve. Egy feladatra 2, illetve 1 pont kapható aszerint, hogy az első vagy csak a második válasz volt a helyes. A T gomb a menühez a HELP billentyű segítségével pedig a bevezetőhöz lehet visszajutni.

A program a RUN/STOP billentyűvel állítható le. A program leállítás után a funkcióbillentyűk átprogramozott állapotban maradnak. Ez a helyzet a RESET gomb megnyomásával szüntethető meg, de ez a programot is törli a memóriából.

```

10 rem sebesseg (gyak.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 fori=1to9
40 readk$(i),t(i),p1(i),p2(i),d1(i),d2(i)
50 next
60 graphic0,1
70 fori=1to7:keyi,"":next:key8,chr$(140)
80 col=10:gosub930
90 printchr$(14);chr$(8)
100 print"          ←"
110 print"      ←          SEBESSEG (gyak.1.)"
120 print" ←"
130 print"☐☐☐A program a  $v = s / t$  összefugges fel-"
140 print"adatokban torteno alkalmazasat gyakorol-tatja."
150 print"☐A valaszokba a meritekegysegeket is be kell irni."
160 print"☐A valaszt a RETURN gomb megnyomasaval kell lezarni."
170 print"☐A program jegyzi az elert eredmenyt."
180 char1,25,24,"TOVABB = ☐T☐"
190 getkeya$:ifa$="t"ora$="T"then200:elseifa$=chr$(140)then60:else
e190
200 scnclr
210 col=7:gosub930
220 char1,5,3,"1. Ut          |"
230 char1,22,4,"|"
240 char1,5,5,"2. Ido          > kiszamitasa"
250 char1,22,6,"|"
260 char1,5,7,"3. Sebesseg      |"
270 char1,5,9,"4. Vegyes feladatok"
280 char1,5,13,"5. Ut          |"
290 char1,22,14,"| kiszamitasa"
300 char1,5,15,"6. Ido          >"
310 char1,22,16,"| szoveg alapjan"
320 char1,5,17,"7. Sebesseg      |"
330 char1,5,19,"8. Vegyes szoveges feladatok"
340 char1,5,23,"☐9. EREDMENYEK"
350 getkeyt$:ift$=chr$(140)then60
360 t=val(t$)
370 ift<1ort>9then350
380 ift=9then1590
390 ift=1theni=1+3*rnd(0)
400 ift=2theni=4+3*rnd(0)
410 ift=3theni=7+3*rnd(0)
420 ift=4thent=int(1+3*rnd(0)):goto390
430 ift=5theni=1+3*rnd(0):t=4
440 ift=6theni=4+3*rnd(0):t=5
450 ift=7theni=7+3*rnd(0):t=6
460 ift=8thent=int(5+3*rnd(0)):goto430
470 :
480 k$=k$(i):p1=p1(i):p2=p2(i):d1=d1(i):d2=d2(i)

```

```

490  scncclr:col=11:gosub930
500  x=int(34+66*rnd(0))*10^(d1-2)
510  y=int(3+7*rnd(0))*10^(d2-1)
520  z=x*y
530  x$=str$(x):ifx<1thenx$=" 0"+right$(x$,len(x$)-1)
540  y$=str$(y):ify<1theny$=" 0"+right$(y$,len(y$)-1)
550  z$=str$(z):ifz<1thenz$=" 0"+right$(z$,len(z$)-1)
560  r=0
570  ontgosub970,1020,1070,1120,1180,1240
580  char1,7,7,"←"
590  a$="":gosub1310
600  ifa$=chr$(140)thenprint"  :r(t)=r(t)-r:goto800
610  h=0
620  ifabs(val(b$)-e)>0.0001thenh=1
630  n=1000000
640  ifabs(val(b$)-e*n)<0.0001andn<>1thenh=2
650  n=n/10:ifn>.0000001then640
660  lb=len(b$):l1=len(m1$):l2=len(m2$)
670  ifright$(b$,l1)<>m1$then690
680  iflb>l1+1thenifval(left$(b$,lb-l1-1))=val(b$)then710:else720
690  ifright$(b$,l2)<>m2$then710
700  iflb>l2+1thenifval(left$(b$,lb-l2-1))=val(b$)then710:else720
710  h=h+4
720  char1,0,10," "
730  if(hand1)=1thenprint"          Szamitasi hiba"
740  if(hand2)=2thenprint"          Nagysagrendi hiba"
750  if(hand4)=4thenprint"          Mertekegyseg hiba"
760  ifh=0then890
770  r=r+1:r(t)=r(t)+1
780  tt=2000:gosub1400
790  ifr=1thenscncclr:goto570
800  char1,0,10,chr$(27)+"q"
810  char1,0,12,chr$(27)+"q"
820  ontgosub1420,1480,1540,1420,1480,1540
830  char1,7,24,"UJ FELADAT =  U  MENU =  M"
840  getkeya$:ifa$=chr$(140)then60
850  ifa$="u"ora$="U"then360
860  ifa$="m"ora$="M"then200
870  goto840
880  :
890  j(t)=j(t)+2-r
900  char1,7,14,"Helyes. "
910  tt=1000:gosub1400
920  goto830
930  :
940  color0,col,1:color1,col,7:color4,col,1
950  return
960  :
970  char1,1,1,"v =" +x$+" m/s"
980  char1,1,3,"t =" +y$+" s"
990  char1,1,5,"-----"
1000 char1,1,7,"s = ?  ":gotol160
1010 return

```

```

1030 char1,1,1,"s="+z$+" m"
1040 char1,1,3,"v="+x$+" m/s"
1040 char1,1,5,"_____"
1050 char1,1,7,"t = ? ":gotol220
1060 return
1070 char1,1,1,"s="+z$+" m"
1080 char1,1,3,"t="+y$+" s"
1090 char1,1,5,"_____"
1100 char1,1,7,"v = ? ":gotol280
1110 return
1120 char1,0,1,k$
1130 char1,0,5,"Mekkora utat tett meg?"
1140 char1,p1,1,x$
1150 char1,p2,1,y$
1160 e=z:m1$="m":m2$="meter"
1170 return
1180 char1,0,1,k$
1190 char1,0,5,"Mennyi a kozben eltelt ido?"
1200 char1,p1,1,z$
1210 char1,p2,1,x$
1220 e=y:m1$="s":m2$="masodperc"
1230 return
1240 char1,0,1,k$
1250 char1,0,5,"Mekkora volt a sebessege?"
1260 char1,p1,1,z$
1270 char1,p2,1,y$
1280 e=x:m1$="m/s":m2$="meter/masodperc"
1290 return
1300 :
1310 b$=""
1320 getkeya$:ifa$=chr$(140)thenreturn
1330 ifa$=chr$(13)andb$<>"thenprint" ":return
1340 ifa$=chr$(20)andb$>"thenprinta$;"← ████";:b$=left$(b$,len(b$)
-1):gotol320
1350 if(a$<="9"anda$>="0")or(a$<="z"anda$>="@")or(a$<="Z"anda$>="
✓")then1380
1360 ifa$="."ora$="*"ora$="/"then1380
1370 gotol320
1380 printa$;"← ████";:b$=b$+a$:iflen(b$)>15thenprint" ":return
1390 gotol320
1400 :
1410 fort=ttto0step-1:next:return
1420 :
1430 char1,7,12,"s = v * t"
1440 char1,7,14,"s="+x$+" m/s *"+y$+" s"
1450 char1,7,16,"s="+z$+" m/s * s"
1460 char1,7,18,"s="+z$+" m"
1470 return
1480 :
1490 char1,7,12,"t = s / v"
1500 char1,7,14,"t="+z$+" m /"+x$+" m/s"
1510 char1,7,16,"t="+y$+" m / m/s"
1520 char1,7,18,"t="+y$+" s"

```

```

1530 return
1540 :
1550 char1,7,12,"v = s / t"
1560 char1,7,14,"v =" +z$+" m /" +y$+" s"
1570 char1,7,16,"v =" +x$+" m/s"
1580 return
1590 :
1600 scnc1r
1610 col=2:gosub930
1620 char1,17,0,"feladat pont eredmeny"
1630 char1,0,3,"Ut"
1640 char1,0,5,"Ido"
1650 char1,0,7,"Sebesseg"
1660 char1,0,11,"Ut (szov.)"
1670 char1,0,13,"Ido (szov.)"
1680 char1,0,15,"Sebesseg (szov.)"
1690 char1,0,19,"OSSZESEN"
1700 fori=1to3
1710 j=j(i):n=j+r(i)
1720 char1,19,1+2*i,"":printusing"###";n/2
1730 char1,26,1+2*i,"":printusing"###";j
1740 ifn=0then1770
1750 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1760 char1,34,1+2*i,"":printusing"### %";j*100/n
1770 color1,2:next
1780 fori=4to6
1790 j=j(i):n=j+r(i)
1800 char1,19,3+2*i,"":printusing"###";n/2
1810 char1,26,3+2*i,"":printusing"###";j
1820 ifn=0then1850
1830 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1840 char1,34,3+2*i,"":printusing"### %";j*100/n
1850 color1,2:next
1860 color1,8,6
1870 j=0:r=0:fori=1to6:j=j+j(i):r=r+r(i):next:n=r+j
1880 char1,19,19,"":printusing"###";n/2
1890 char1,26,19,"":printusing"###";j
1900 ifn=0then1920
1910 char1,34,19,"":printusing"### %";j*100/n
1920 color1,2
1930 gotol80
1940 data"Egy repulo ... s ideig .. m/s sebesseg- igel repult."
1950 data1,22,10,2,3
1960 data"Egy hajo ... masodpercig ... m/s sebesseggel ha
ladt."
1970 data1,24,8,1,3
1980 data"Egy test masodpercen at m/s sebesseggel mo
zgott."
1990 data1,28,8,1,4
2000 data"Egy hajo m/s sebesseggel m utat tett meg."
2010 data2,28,8,1,3

```



2020 data"Egy helikopter m utat m/s sebességgel te  
 tt meg."  
 2030 data2,14,26,2,2  
 2040 data"Egy nyul m utat m/s sebességgel tett meg."  
 2050 data2,8,19,1,2  
 2060 data"Egy repülőgépet m utat tett meg másodperc alatt."  
 2070 data3,13,35,3,2  
 2080 data"Egy hangya s alatt m utat tett meg."  
 2090 data3,20,10,-1,1  
 2100 data"Egy vonat m utat tett meg s alatt."  
 2110 data3,9,29,2,1

	osztály	csoport	egyéni:
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 185 s (39 mp.)

Speciális karakter: nincs

A program a sebesség fogalmának elmélyítéséhez készült. A  $v = s / t$  képletben szereplő három mennyiség közül a gép 2 - 2 mennyiség nagyságrendi viszonyait megadja, és ennek ismeretében kell a tanulónak a hiányzó mennyiségeket összehasonlítani. (Pl. a gép megadja, hogy a megtett utak egyenlők, a mozgás időtartama viszont az egyik esetben kétszer nagyobb. Ekkor a sebességeket kell összehasonlítani.)

Elindítás után egy rövid tájékoztatót ad a program. Ezt elolvasva a "T" gomb lenyomásával lehet továbblépni. Ezután történik a nehézségi fokozat kiválasztása. A felsorolt lehetőségek közül a megfelelő szám lenyomásával lehet választani. Itt van mód arra, hogy az addig elért eredményt a 4 gomb lenyomásával kiírassuk a képernyőre. A HELP billentyű lenyomásával a bevezetőhöz lehet visszamenni.

A nehézségi fokozat meghatározása után az összehasonlítandó mennyiségeket kell kiválasztani. Itt is a megfelelő szám lenyomásával lehet választani. Ezután a program a választásokat figyelembevéve véletlenszerű értéket ad 2 - 2 mennyiségnek, és rákérdez a hiányzó mennyiségek összehasonlításának eredményére. A választ a > = < és a ? gombok segítségével lehet beírni. A ? gombot akkor kell lenyomni, ha a kérdésre nem adható egyértelmű válasz. A HELP billentyű megnyomására a program megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az eredménybe. A válasz beírása után, ha az hibás volt, a program az erre utaló üzenet kiírása után megadja a helyes választ, ellenkező esetben csak a REMEK felirat jelenik meg a képernyőn. Ezután az U gomb lenyomásával új feladatot kérhetünk, az M gomb lenyomásával pedig a menühöz (nehézségi fokozat kiválasztása és eredmények) térhetünk vissza. A HELP billentyű megnyomása ebben az esetben a bevezetőhöz való visszatérést eredményezi.

Az eredmények kiírását az említett módon a menü alapján, a 4 gomb

lenyomásával) kérhetjük. A program nehézségi fokozat és az összehasonlítandó mennyiség szerint csoportosítva, valamint összesítve százalékban adja meg az elért eredményt. Az eredmények kiírása után a T gombbal újra a menühez lehet visszajutni, míg a HELP billentyű lenyomásával a bevezető ismertetéshez léphetünk vissza. A program a RUN/STOP billentyűvel állítható le. A program leállítása után a funkcióbillentyűk átállított állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez a program törlését eredményezi.

```

10 rem sebesseg (gyak 2.)
20 rem zatonyi sandor bekescsaba 1986.
30 scnclr:restore
40 fori=1to7:keyi, "":next:key8,chr$(140)
50 fok$(1)="Konnyu ":t$(1)="Ut"
60 fok$(2)="Kozepes":t$(2)="Ido"
70 fok$(3)="Nehez  ":t$(3)="Sebesseg"
80 t$(4)="VEGYES"
90 col=11:gosub2020
100 printchr$(14);chr$(8)
110 print"                ←"
120 print"          ←          SEBESSEG (gyak 2.)"
130 print" ←"
140 print"☐☐☐A program a  $v = s / t$  összefugges"
150 print"elemzesehez keszult."
160 print"☐A megadott adatok alapjan ossze kell"
170 print"hasonlitani a ket mennyiseget."
180 print"☐Az osszehasonlitasi eredmenyet a < = >"
190 print"ombokkal lehet beirni."
200 print"☐Ha nem adható egyertelmu valasz, akkor"
210 print"a ? gombot kell megnyomni."
220 char1,25,24,"TOVABB = ☐T☐"
230 getkeya$:ifa$=chr$(140)then30
240 ifa$<"t"anda$<"T"then230
250 scnclr
260 col=10:gosub2020
270 char1,5,0,"NEHEZSEGI FOKOZAT VALASZTAS"
280 char1,14,7,"1. "+fok$(1)
290 char1,14,10,"2. "+fok$(2)
300 char1,14,13,"3. "+fok$(3)
310 char1,14,18,"4. EREDMENYEK"
320 char1,2,24,"A szam lenyomasaval lehet valasztani."
330 getkeya$:ifa$=chr$(140)then30
340 ifa$="4"then1740
350 ifa$<"1"ora$>"3"then330
360 f=val(a$)
370 scnclr
380 col=10:gosub2020
390 gosub2050
400 char1,4,7,"AZ OSSZEHASONLITANDO MENNYISEG"
410 char1,16,12,"1. "+t$(1)
420 char1,16,14,"2. "+t$(2)
430 char1,16,16,"3. "+t$(3)
440 char1,16,20,"4. "+t$(4)

```

```

450 char1,2,24,"A.szam lenyomasaval lehet valasztani."
460 getkey$:ift$=chr$(140)then30
470 ift$<"1"ort$>"4"then460
480 t=val(t$):ift=4thent=1+3*rnd(0)
490 scnclr
500 col=11:gosub2020
510 gosub2050
520 x1=int(1+9*rnd(0)):x2=int(1+9*rnd(0))
530 y1=int(1+9*rnd(0)):y2=int(1+9*rnd(0))
540 r=rnd(0)
550 ifr<.3then580
560 ifr<.65thenx1=x2
570 ifr>=.65theny1=y2
580 ontgosub810,1120,1430
590 char1,8,5,x1$:char1,27,5,x2$
600 char1,8,9,y1$:char1,27,9,y2$
610 char1,7,11,"_____ "
620 char1,8,13,z1$:char1,27,13,z2$
630 char1,19,13," 37 11"
640 getkeya$:ifa$=chr$(140)then740
650 ifa$=","ora$="."ora$="/"thena$=chr$(asc(a$)+16)
660 ifa$<>"<"anda$<>"="anda$<>">"anda$<>"?"then640:elsechar1,19,1
3,a$
670 n(f,t)=n(f,t)+1
680 ifa$=e$then690:else720
690 j(f,t)=j(f,t)+1
700 char1,17,18,"REMEK"
710 goto760
720 char1,17,16,"ROSSZ"
730 fort=0to300:next
740 char1,11,20,"A helyes valasz: 37 11"+e$+" "
750 fort=0to900:next
760 char1,5,24,"UJ FELADAT = U      MENU = M"
770 getkeya$:ifa$=chr$(140)then30
780 ifa$="m"ora$="M"then250
790 ifa$="u"ora$="U"then480
800 goto770
810 :
820 z1$=" s 1":z2$=" s 2"
830 onfgoto840,890,1000
840 :
850 x1$=str$(x1)+"m/s":x2$=str$(x2)+"m/s"
860 y1$=str$(y1)+"s":y2$=str$(y2)+"s"
870 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
"
880 return
890 :
900 x1$=str$(x1)+"v":x2$=str$(x2)+"v"
910 ifx1=1thenx1$=" v"
920 ifx2=1thenx2$=" v"
930 ifx1=x2thenx1$=" v":x2$=x1$
940 y1$=str$(y1)+"t":y2$=str$(y2)+"t"
950 ify1=1theny1$=" t"

```

```

960 ify2=1theny2$=" t"
970 ify1=y2theny1$=" t":y2$=y1$
980 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
"
990 return
1000 :
1010 x1$=" v[1]":x2$=" v[2]"
1020 y1$=" t[1]":y2$=" t[2]"
1030 ifx1=x2andyl=y2thene$="=":gotol070
1040 ifx1<=x2andyl<=y2thene$="<":gotol070
1050 ifx1>=x2andyl>=y2thene$=">":gotol070
1060 e$="?"
1070 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1080 char1,19,5,r$
1090 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1100 char1,19,9,r$
1110 return
1120 :
1130 z1$=" t[1]":z2$=" t[2]"
1140 onfgotol160,1200,1310
1150 :
1160 x1$=str$(x1)+"m":x2$=str$(x2)+"m"
1170 y1$=str$(y1)+"m/s":y2$=str$(y2)+"m/s"
1180 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$=">"
"
1190 return
1200 :
1210 x1$=str$(x1)+"s":x2$=str$(x2)+"s"
1220 ifx1=1thenx1$=" s"
1230 ifx2=1thenx2$=" s"
1240 ifx1=x2thenx1$=" s":x2$=x1$
1250 y1$=str$(y1)+"v":y2$=str$(y2)+"v"
1260 ify1=1theny1$=" v"
1270 ify2=1theny2$=" v"
1280 ify1=y2theny1$=" v":y2$=y1$
1290 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$=">"
"
1300 return
1310 :
1320 x1$=" s[1]":x2$=" s[2]"
1330 y1$=" v[1]":y2$=" v[2]"
1340 ifx1=x2andyl=y2thene$="=":gotol380
1350 ifx1<=x2andyl>=y2thene$="<":gotol380
1360 ifx1>=x2andyl<=y2thene$=">":gotol380
1370 e$="?"
1380 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1390 char1,19,5,r$
1400 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1410 char1,19,9,r$
1420 return
1430 :
1440 z1$=" v[1]":z2$=" v[2]"
1450 onfgotol460,1510,1620

```

```

1460 :
1470 x1$=str$(x1)+"m":x2$=str$(x2)+"m"
1480 y1$=str$(y1)+"s":y2$=str$(y2)+"s"
1490 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1500 return
1510 :
1520 x1$=str$(x1)+"s":x2$=str$(x2)+"s"
1530 ifx1=1thenx1$=" s"
1540 ifx2=1thenx2$=" s"
1550 ifx1=x2thenx1$=" s":x2$=x1$
1560 y1$=str$(y1)+"t":y2$=str$(y2)+"t"
1570 ify1=1theny1$=" t"
1580 ify2=1theny2$=" t"
1590 ify1=y2theny1$=" t":y2$=y1$
1600 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1610 return
1620 :
1630 x1$=" s□1":x2$=" s□2"
1640 y1$=" t□1":y2$=" t□2"
1650 ifx1=x2andyl=y2thene$="=":gotol690
1660 ifx1<=x2andyl>=y2thene$="<":gotol690
1670 ifx1>=x2andyl<=y2thene$=">":gotol690
1680 e$="?"
1690 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1700 char1,19,5,r$
1710 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1720 char1,19,9,r$
1730 return
1740 :
1750 col=15:gosub2020
1760 scnclr
1770 char1,15,0,"EREDMENYEK"
1780 char1,14,5,"1.      2.      3.      OSSZ."
1790 char1,0,9,t$(1)
1800 char1,0,12,t$(2)
1810 char1,0,15,t$(3)
1820 char1,0,19,"OSSZESEN"
1830 fort=1to3
1840 n(4,t)=0:j(4,t)=0
1850 forf=1to3
1860 ifn(f,t)>0thene(f,t)=100*j(f,t)/n(f,t)
1870 n(4,t)=n(4,t)+n(f,t)
1880 j(4,t)=j(4,t)+j(f,t)
1890 next
1900 ifn(4,t)>0thene(4,t)=100*j(4,t)/n(4,t)
1910 next
1920 forf=1to4
1930 n(f,4)=0:j(f,4)=0
1940 fort=1to3
1950 ifn(f,t)>0thenchar1,5+7*f,6+3*t,"":printusing"###%";e(f,t)

```

```

1960 n(f,4)=n(f,4)+n(f,t)
1970 j(f,4)=j(f,4)+j(f.t)
1980 next
1990 ifn(f,4)>0thenchar1,5+7*f,19,"":printusing"###%";100*j(f,4)/
n(f,4)
2000 next
2010 goto220
2020 :
2030 color0,col,1:color1,col,7:color4,col,1
2040 return
2050 :
2060 char1,0,0," "
2070 char1,0,1," " Fokozat: "+fok$(f)+" "
2080 char1,0,2," " "
2090 return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 95 s (31 mp.)

Speciális karakter: nincs

A program a sebesség fogalmának elmélyítéséhez használható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10 féle testre (gyalogos, kerékpáros, Trabant, helikopter, Boeing 707, hang levegőben, Tu 144, hang vasban, űrhajó, Föld) vonatkozó adatok szerepelnek. Az első oszlopban a test neve, a másodikban az adott idő alatt megtett út, a harmadikban az idő (kezdeti értéke 10 s), az utolsóban pedig a sebesség értéke található.

A gyalogos útjára vonatkozó adat előtt álló > jel a kurzormozgató billentyűkkel (nyilakkal) bármelyik test út- vagy időadata elé vihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban (sötét alapon világos betűk) jelenik meg a képernyőn.

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Például a ↓ gomb kétszeri és a → gomb egyszeri megnyomása után a Trabantra vonatkozó időadat módosítható. Ha ekkor 100-at írunk be, és megnyomjuk a RETURN billentyűt, a 3. sorban a Trabant útjánál 3000, az időnél 10.0 és a sebességnél 30.0 fog megjelenni. Ezzel az eljárással egy kiválasztott test esetén az út és az idő közti egyenes arányosság tanulmányozható. (A program egyetlen mozgást tételez fel.)

Lehetőség van arra is, hogy valamennyi anyag út, illetve időadatát egyszerre változtassuk. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja. Ezzel lehetőség nyílik azonos időtartamú, illetve azonos úton végbemenő mozgások út- (illetve



időadatainak) az összehasonlítására.

A programban szereplő testek a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni a test nevét, majd az idézőjelen kívülre, a névtől egy vesszővel elválasztva a sebességét. Figyelem, a tizedesvessző helyett tizedespontot kell írni! Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A program leállítása után a funkcióbillentyűk továbbra is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ez az eljárás a programot is törli a memóriából.

```
10 rem sebesseg (tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 fori=2to7:keyi,"":next
40 key8,chr$(140)
50 scnclr:color0,16,2:color1,2,7:color4,16,2
60 printchr$(14);chr$(8)
70 print"  _____ X"
80 print"    _A_____/"
90 print"      (____)JJJ      SEBESSEG (tabl.1.)"
100 print"        J  L"
110 print"  A programmal 10 kulonbozo test sebessege"
120 print"tanulmanyozhato."
130 print"  A > jel a billentyuzeten talalhato"
140 print"nyilakkal a megfelelo helyre viheto."
150 print"  A kivant ertekek beirasa utan a RETURN"
160 print"gomb megnyomasara a program atirja a"
170 print"tablazat adatait."
180 print"  A beirt ertekek legfeljebb 4 karakterbol"
190 print"allhat."
200 print"  Az ESC billentyu megnyomasa utan az"
210 print"egeszesz oszlop erteke atirhato."
220 char1,25,24,"TOVABB = T"
230 fori=0to9:readnev$(i),z(i):next
240 getkeya$:ifa$=chr$(140)thenrun
250 ifa$<>"t"anda$<>"T"then240
260 color0,10,2:color1,2,7:color4,10,2
270 fori=0to9:y(i)=10:x(i)=y(i)*z(i):next
280 p=1:q=0
290 scnclr
300 print"          UT          IDO          SEBESSEG"
310 print"          (m)          (s)          (m/s)"
320 fori=0to9:gosub500:next
330 b$=""
340 char1,12*p-2,4+2*q,">"
350 getkeya$:ifa$=chr$(140)thenrun
360 ifa$="|"andp=2andb$=""thenp=1:print" ":goto330
370 ifa$="|"andp=1andb$=""thenp=2:print" ":goto330
380 ifa$="|"andq>0andb$=""thenq=q-1:print" ":goto330
390 ifa$="|"andq<9andb$=""thenq=q+1:print" ":goto330
400 ifa$="|"andb$=""thenprint" ":p=1:q=0:goto330
```

```

410 ifa$="☒"andb$=""thenp=2:fori=0to9:y(i)=10:gosub500:next:p=1:g
oto330
420 ifa$=chr$(27)andb$=""thengosub590:goto330
430 if(a$<"0"ora$>"9")anda$<>". "then350
440 i=q
450 print"          "
460 char1,12*p-2,4+2*i,"> "
470 gosub680
480 gosub750
490 goto330
500 :
510 ifp=1theny(i)=x(i)/z(i)
520 ifp=2thenx(i)=y(i)*z(i)
530 char1,0,4+2*i,nev$(i)
540 char1,10,4+2*i," "
550 printusing"##### ";x(i);
560 printusing"#####.# ";y(i);
570 printusing"#####.#";z(i);
580 return
590 :
600 print"█ ":char1,12*p-2,24,"> "
610 gosub680
620 fori=0to9:gosub750:next
630 char1,12*p-2,24," "
640 q=0
650 return
660 :
670 getkeya$
680 ifa$=chr$(140)thenrun
690 ifa$=chr$(13)andb$>" "then740
700 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):print"█ █";:g
oto670
710 if(a$>"9"ora$<"0")anda$<>". "then670
720 b$=b$+a$
730 print"█"+a$;" ";:iflen(b$)<4then670
740 return
750 :
760 b=val(b$)
770 ifp=1thenx(i)=b:elsey(i)=b
780 gosub500
790 return
800 :
810 data"Gyalogos",1.4
820 data"Kerekparos",5
830 data"Trabant",30
840 data"Helikopter",80
850 data"Boeing 707",280
860 data"Hang(lev.)",340
870 data"TU 144",690
880 data"Hang (vas)",5100
890 data"Urhajo",7900
900 data"Fold",29800

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 185 s (38 mp.)

Speciális karakter: nincs

A program a  $v = s / t$  összefüggés vizsgálatához készült. A képernyőn szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább a menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, aminek az értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésközt 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységet nem kell beírni, mert a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól. Ezek után három lehetőség közül lehet választani. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN billentyű megnyomásával az előzőleg használt értéket megtartsuk. A program futása a RUN/STOP billentyű megnyomásával szakítható meg. A funkcióbillentyűk azonban a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez egyúttal a program törlését is jelenti.

10 rem sebesség (tabl.2.)

20 rem zatonyi sandor bekescsaba 1986.

60 scnc1r

160 fori=1to7:keyi, "" :next:key8, chr\$(140)

170 f1\$=" UT IDO SEBESSEG "

180 f2\$=" (m) (s) (m/s)

190 v1\$="UT"

200 v2\$="IDO"

210 v3\$="SEBESSEG"

220 v\$="-----"

230 sp\$=" "

240 col=9:gosub1990

250 printchr\$(14);chr\$(8);

260 print" ←"

270 print" ← SEBESSEG (tabl.2.)"

280 print" ←"

290 print"☐☐☐A program a  $v = s / t$  összefüggés elem-

300 print"zésehez készült."

310 print"☐A három mennyiség közül egyet állandó"

320 print"értéknek választva egy másik mennyiség"

330 print"értéke szabadon változtatható."

340 print"☐A program ezek figyelembevételével"

350 print"kiszámítja a harmadik változó értékeit."

370 print"☐A CLEAR/HOME megnyomására új értékek"

380 print"adhatók az egyes mennyiségeknek, a"

390 print"SHIFT és CLEAR/HOME együttes lenyomása"

400 print"után pedig az egyes mennyiségek szerepe"

410 print"változtatható meg."

420 char1,25,24,"TOVABB = ☐T☐"

430 getkeya\$:ifa\$=chr\$(140)thenrun

440 ifa\$<>"t"anda\$<>"T"then430

450 col=10:gosub1990

460 all=50:kezd=10:lep=1

470 print"☐"+sp\$+" A VIZSGALT ESET KIVALASZTASA "+sp\$

480 char1,7,5,"FUGGO FUGGETLEN "

490 char1,18,6,"ALLANDO"

500 char1,6,7,"VALTOZO VALTOZO "

510 char1,0,8,v\$

520 char1,0,9," 1. UT IDO SEBESSEG "+sp\$

530 char1,0,11," 2. UT SEBESSEG IDO "+v\$

540 char1,0,13," 3. IDO UT SEBESSEG "+sp\$

550 char1,0,15," 4. IDO SEBESSEG UT "+v\$

560 char1,0,17," 5. SEBESSEG UT IDO "+sp\$

570 char1,0,19," 6. SEBESSEG IDO UT "+v\$

580 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET VALASZTA

NI."

```

590 getkeyts$:ift$=chr$(140)thenrun
600 ift$<"1"ort$>"6"then590
610 col=15:gosub1990
620 print"☐";sp$;"          ERTEKADAS          ";sp$
630 char1,2,5,"FUGGO VALTOZO      :"
640 char1,2,7,"Allando mennyiseg  :"
650 char1,2,9,"Fuggetlen valtozo  :"
660 onval(t$)goto670,850,1030,1210,1390,1570
670 :
680 char1,22,5,v1$
690 char1,22,7,v2$
700 char1,22,9,v3$
710 char1,0,11,v$
720 char1,0,14,"Az ido erteke (s)  : "
730 a=all:p=20:q=14:gosub1750:all=a
740 char1,0,17,"A sebesseg"
750 char1,5,19,"- kezdoerteke (m/s) : "
760 a=kezd:p=27:q=19:gosub1750:kezd=a
770 char1,5,21,"- lepeskoz (m/s)  : "
780 a=lep:p=24:q=21:gosub1750:lep=a
790 col=9:gosub1990
800 char1,5,1,"☐—☐"
810 forj=0to9
820 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
830 next
840 goto1880
850 :
860 char1,22,5,v1$
870 char1,22,7,v3$
880 char1,22,9,v2$
890 char1,0,11,v$
900 char1,0,14,"A sebesseg erteke (m/s) : "
910 a=all:p=26:q=14:gosub1750:all=a
920 char1,0,17,"Az ido"
930 char1,5,19,"- kezdoerteke (s)  : "
940 a=kezd:p=25:q=19:gosub1750:kezd=a
950 char1,5,21,"- lepeskoz (s)  : "
960 a=lep:p=22:q=21:gosub1750:lep=a
970 col=9:gosub1990
980 char1,5,1,"☐—☐"
990 forj=0to9
1000 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
1010 next
1020 goto1880
1030 :
1040 char1,22,5,v2$
1050 char1,22,7,v1$
1060 char1,22,9,v3$
1070 char1,0,11,v$
1080 char1,0,14,"Az ut erteke (m)  : "
1090 a=all:p=19:q=14:gosub1750:all=a
1100 char1,0,17,"A sebesseg"
1110 char1,5,19,"- kezdoerteke (m/s)  : "

```

```

1120 a=kezd:p=27:q=19:gosubl750:kezd=a
1130 char1,5,21,"- lepeskoz (m/s) : "
1140 a=lep:p=24:q=21:gosubl750:lep=a
1150 col=9:gosubl990
1160 char1,19,1,"  ────  "
1170 forj=0to9
1180 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1190 next
1200 goto1880
1210 :
1220 char1,22,5,v2$
1230 char1,22,7,v3$
1240 char1,22,9,v1$
1250 char1,0,11,v$
1260 char1,0,14,"A sebesseg erteke (m/s) : "
1270 a=all:p=30:q=14:gosubl750:all=a
1280 char1,0,17,"Az ut"
1290 char1,5,19,"- kezdoerteke (m) : "
1300 a=kezd:p=25:q=19:gosubl750:kezd=a
1310 char1,5,21,"- lepeskoz (m) : "
1320 a=lep:p=22:q=21:gosubl750:lep=a
1330 col=9:gosubl990
1340 char1,19,1,"  ────  "
1350 forj=0to9
1360 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1370 next
1380 goto1880
1390 :
1400 char1,22,5,v3$
1410 char1,22,7,v1$
1420 char1,22,9,v2$
1430 char1,0,11,v$
1440 char1,0,14,"Az ut erteke (m) : "
1450 a=all:p=19:q=14:gosubl750:all=a
1460 char1,0,17,"Az ido"
1470 char1,5,19,"- kezdoerteke (s) : "
1480 a=kezd:p=25:q=19:gosubl750:kezd=a
1490 char1,5,21,"- lepeskoz (s) : "
1500 a=lep:p=22:q=21:gosubl750:lep=a
1510 col=9:gosubl990
1520 char1,29,1,"  ────────────  "
1530 forj=0to9
1540 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1550 next
1560 goto1880
1570 :
1580 char1,22,5,v3$
1590 char1,22,7,v2$
1600 char1,22,9,v1$
1610 char1,0,11,v$
1620 char1,0,14,"Az ido erteke (s) : "
1630 a=all:p=20:q=14:gosubl750:all=a
1640 char1,0,17,"Az ut"

```

```

1650 char1,5,19,"- kezdoerteke (m) : "
1660 a=kezd:p=25:q=19:gosub1750:kezd=a
1670 char1,5,21,"- lepeskoz (m) : "
1680 a=lep:p=22:q=21:gosub1750:lep=a
1690 col=9:gosub1990
1700 char1,29,1," ┌───────────┐"
1710 forj=0to9
1720 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1730 next
1740 goto1880
1750 :
1760 b$="":char1,p,q," "
1770 print" ←[ ]";:getkeya$
1780 ifa$=chr$(13)then1820
1790 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$+a$
1800 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1810 iflen(b$)<6then1770
1820 ifb$=" "then1850
1830 ifval(b$)<0.01thenchar1,p,q,sp$:goto1750
1840 a=val(b$)
1850 char1,p,q,str$(a)+sp$
1860 ifa<1thenchar1,p,q,"0"
1870 return
1880 .
1890 char1,0,0,f1$:char1,0,3,f2$
1900 forj=0to9
1910 char1,0,6+2*j," "
1920 printusing"#####.#";x(j);
1930 printusing"#####.#";y(j);
1940 printusing"#####.#";z(j);
1950 next
1960 getkeya$:ifa$="█"then610:elseifa$=""then450:elseifa$=chr$(14
0)thenrun
1970 goto1960
1980 stop
1990 :
2000 scnclr
2010 color0,col,6:color1,col,1:color4,col,6
2020 return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 85 s (29 mp.)

Speciális karakter: nincs

A program segítségével a változó sebességű mozgások vizsgálhatók, és alkalmazása hozzájárul az átlagsebesség fogalmának jobb megértéséhez. Az indításkor megjelenő ismertető elolvasása után választani kell, hogy a mozgásról rajzoljon-e nyomképet a program. A választás az I illetve az N gombok egyikének megnyomásával történik.

Ezt követően a program felrajzolja az út - idő grafikonhoz szükséges koordinátarendszert, a testet jelző kis négyzetet és kiírja a START=S feliratot. Az S gomb megnyomására a test véletlenszerű pillanatnyi sebességgel halad. A program a test mozgása közben megrajzolja az út - idő grafikon néhány pontját, és ha kértük, akkor megjeleníti az ezekhez a helyzetekhez tartozó nyomképeket is.

Ha a test elérte a képernyő tetejét, a megjelenő feliratoknak megfelelően több lehetőség közül is választhatunk:

Az U gomb megnyomásával a mozgás újra megismételhető. A T gomb hatására a program továbblép egy másik mozgás vizsgálatához. Az új mozgás út - idő grafikonja az előzőleg használt koordinátarendszerben, a korábban rajzoltak törlése nélkül jelenik meg. Ezáltal a különböző mozgások grafikonjai együtt is tanulmányozhatók. Amennyiben a meglévő grafikonokat törölni akarjuk, a SHIFT és a CLEAR/HOME gombokat kell együtt lenyomni.

A G billentyű megnyomásával a berajzolt pontokhoz görbét illeszt a program. A görbe rajzolásával egyidejűleg kiírja a mozgás kezdete óta megtett út, az eltelt idő és az átlagsebesség értékét is. A görbe megrajzolása után már csak az ÚJRA = U és a TOVÁBB = T feliratok jelennek meg.

A HELP billentyű megnyomásával itt is a bevezetőhöz lehet visszatérni.

A program a RUN/STOP billentyűvel állítható meg. A leállítás lehetőleg szöveges üzemmódban történjen, mert különben a READY üzenet és a villogó kurzor csak a GRAPHIC 0 parancs "vakon"





	osztály	csoport	egyéni
új ismeret	○	●	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 85 s (29 mp.)

Speciális karakter: nincs

A programmal az egyenletes változómozgás vizsgálható, és használata segíthet a gyorsulás fogalmának elmélyítésében is. Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva választani kell, hogy a mozgás nyomképét megrajzolja-e a program. Ehhez csupán az I, illetve az N gombok egyikét kell megnyomni. Ennek megtörténtével a program felrajzolja az út - idő grafikon koordinátarendszerét, a testet jelző kis négyzetet és megjelenik a START = S felirat. Az S gomb megnyomására a test véletlenszerűen választott gyorsulással egyenletesen változó mozgást végez. A program a mozgással egyidejűleg berajzolja a koordinátarendszerbe az út - idő grafikon néhány pontját, és a korábbi választásunknak megfelelően a hozzájuk tartozó nyomképet is.

A mozgás végeztével a képernyőn megjelenő feliratok alapján több lehetőség közül is választhatunk:

Az U gomb megnyomásával a mozgás újra megismételhető. A T gomb lenyomásával továbbléphetünk egy más gyorsulással történő mozgás vizsgálatához. Az újabb mozgás út-idő grafikonja az előző koordinátarendszerben, a korábban megrajzolt grafikonokkal együtt jelenik meg. Így a különböző gyorsulással végbemenő mozgások grafikonjai közvetlenül is összehasonlíthatók. Ha a berajzolt grafikonokat törölni kívánjuk, a SHIFT és a CLEAR/HOME gombokat kell együttesen lenyomni.

A G megnyomásával a koordinátarendszerbe előzőleg bejelölt pontokhoz görbét (parabolát) illeszt a program. A görbe megrajzolásával egyidejűleg az út, az idő és a pillanatnyi sebesség értékét is folyamatosan kiírja a program. A grafikon kirajzolása után már csak az ÚJRA = U és a TOVÁBB = T felirat jelenik meg.

A HELP billentyű megnyomása ebben a programban is az ismertetőhöz való visszatérést eredményezi.

A program a RUN/STOP billentyűvel állítható le. Ha a leállítás grafikus üzemmódban történne és ezért a READY felirat és a villogó kurzor nem jelenne meg, a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN billentyű megnyomásával visszatérhetünk a normál, szöveges üzemmódbhoz. A funkcióbillentyűk a program leállítása után is átprogramozott állapotban maradnak. A RESET gomb megnyomása alaphelyzetbe állítja őket, de ez az eljárás a program törlésével jár.

```

10 rem gyorsulo mozgás
20 rem zatonyi sandor bekescsaba 1986.
30 deffns(t)=int(180.5-a*t*t):fort=1to7:keyt,"":next:key8,chr$(8)
:poke740,212
40 sp$="          ":s$="START=S":t$="TOVABB=T":u$="UJRA=U":g$="GR
AFIKON=G"
50 color0,9,2:color4,9,2:color1,2,7:graphic0,1:printchr$(14);chr$
(8);
60 print"
70 print"
80 print"testet es felrajzolja az ut - ido          grafikon nehany
pontjat."
90 print"Ezekhez a G gomb megnyomasara grafikont rajzol."
100 print"A mozgás az U gomb megnyomasaval ujra"
110 print"indul, a T hatásara pedig másik mozgás vizsgálható."
120 char1,13,24,"NYOMKEPET RAJZOLJAK? (I/N)"
130 geta$:ifa$="i"ora$="I"thenj=1:elseifa$="n"ora$="N"thenj=0:els
e130
140 graphic1,1
150 h=180:fi=270:locate320,180:gosub380
160 h=7.1:fi=45:locate135,5:gosub380:locate314,175:gosub380
170 fort=20to180step20:draw1,137,ttol140,t:draw1,120+t,180to120+t,
183:next
180 char1,18,1,"s(m)":char1,35,21,"t(s)"
190 char1,17,24,"0          5"
200 char1,15,22,"0":char1,14,10,"50"
210 a=int(15*rnd(0)+1)/500:g=0
220 t=0:s=180:n=1:gosub390:char1,0,0,s$:getkeya$:ifa$=chr$(8)then
run
230 ifa$<>"s"anda$<>"S"then220:elsechar1,0,0,sp$
240 dountils<0:n=j:gosub390:t=t+20:s=fns(t):n=1:gosub390:draw1,14
0+t,s:loop
250 ifg=0thenchar1,0,9,g$
260 char1,0,11,u$:char1,0,13,t$
270 getkeya$:ifa$=chr$(8)thenrun:elseif(a$="g"ora$="G")andg=0then
300
280 ifa$="u"then140:elseifa$="u"ora$="U"thengosub400:goto220
290 ifa$="t"ora$="T"thengosub400:goto210:else270
300 fort=9to24:char1,0,t,sp$:next:t=5:draw1,140,180
310 char1,0,20,"s =          m"
320 char1,0,22,"t =          s"
330 char1,0,24,"v =          m/s"

```

```
340 dountilt>180orfns(t)<0:drawto140+t, fns(t)
350 c=0:m=a*t*t/2:gosub410:c=2:m=t/20:gosub410:c=4:m=a*t*20:gosub
410
360 char1,0,13,"TOVABB=G":geta$:ifa$="g"ora$="G"then370:else360
370 t=t+5:loop:g=1:goto260
380 drawtoh;fitch;fi+90:fi=fi+90:return
390 boxn,90,s,100,s-10:return
400 fort=0to24:char1,0,t,sp$+" " :next:return
410 a$=str$(m)+sp$:ifm<1thena$=" 0"+mid$(a$,2,3)
420 char1,3,20+c,left$(a$,5):return
```

	osztály	csoport	egyéni
új ismeret	○	●	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 80 s (29 mp.)

Speciális karakter: nincs

A programmal különböző meredekségű lejtőkön létrejövő mozgás vizsgálható. Az indítás után megjelenő ismertető elolvasása után a lejtő vízszintessel bezárt szögét kell megadni. Ennek értéke 1 és 45 fok közötti (egész szám) lehet. Ezt követően választani kell, hogy nyomkép maradjon-e a mozgás egyes fázisairól, vagy sem. Ehhez az I illetve az N gombok valamelyikét kell lenyomni. Ennek megtörténte után a program felrajzolja az adott szögnek megfelelő lejtőt, az út - idő grafikonhoz szükséges koordinátarendszert, és kiírja a lejtő hajlásszögének értékét. A mozgás a képernyőn megjelenő feliratnak megfelelően az S gomb (Start) megnyomására indul. A mozgással egyidőben a program berajzolja az út-idő grafikon egyes pontjait, és ha kértük, az ezekhez tartozó nyomképeket is rögzíti. A képernyő bal felső sarkában látható mezőben megjelenő szám a mozgás során kirajzolt nyomképek számát, és ezzel együtt az eltelt időt jelzi.

A mozgás végén a képernyőn megjelenő feliratnak megfelelően az U gomb megnyomásával a mozgás újra megismételhető, a T gombbal (vagy a HELP billentyű segítségével) a bevezetőhöz mehetünk vissza, és másik hajlásszöget adhatunk meg. A G gomb segítségével a koordinátarendszerben előzőleg felrajzolt pontokhoz görbét (parabolát) illeszt a program. A görbe megrajzolása után már csak az ÚJRA = U és a TOVABB = T felirat jelenik meg és csak e két lehetőség közül választhatunk.

A program futása a RUN/STOP billentyűvel állítható meg. Ha a leállítás grafikus üzemmódban történt, a READY felirat és a villogó kurzor nem jelenik meg. Ekkor a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN billentyű lenyomásával térhetünk vissza a normál üzemmódba. A leállítás után a funkcióbillentyűk továbbra is átprogramozott állapotban maradnak. Ez az állapot a RESET gomb megnyomásával megszüntethető, de ez az eljárás a programot is törli.

```

10 rem mozgas lejtón
20 rem zatonyi sándor bekescsaba 1986.
30 fi=10:printchr$(14);chr$(8):fori=1to7:keyi,"":next:key8,chr$(1
40):poke740,212
40 s$="START = S":u$="UJRA = U":t$="TOVABB = T":g$="GORBE = G":sp
$="
50 graphic0,1:color0,9,2:color4,9,2:color1,9,7:a$="MOZGAS LEJTÓN"
60 fori=0to5:char1,3+i*3,i,mid$(a$,i+1,1):char1,20+i*3,i,mid$(a$,
i+8,1):next
70 print:print"☐☐☐A programmal különbozo meredeksegu"
80 print"lejtókon letrejo vo mozgas vizsgalható."
90 print"☐☐☐A lejtónek a vízszintessel bezart szöge 1 es 45 fok köz
t lehet."
100 print"☐☐☐A program a mozgással egyidoben fel-"
110 print"rajzolja az út-ido grafikon megfelelo pontjait."
120 print"☐☐☐Ezekhez a G megnyomasara gorbet illeszt."
130 char1,0,22,"Hany fokos a lejtó hajlasszöge? ":b$=""
140 print" ☐←☐☐☐";:getkeya$:ifa$=chr$(140)thenrun:elseifa$=chr$(1
3)then190
150 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):gotol80
160 ifa$="0"andb$>"0"thenb$=b+a$:gotol80
170 ifa$<"1"orab$>"9"then140:elseb$=b+a$
180 printa$;:iflen(b$)<3then140
190 fi=val(b$):iffi>45orfi=0thenchar1,32,22,sp$:gotol30
200 char1,0,22,"Nyomkep maradjon? (i/n)" +chr$(27)+"q"
210 getkeya$:ifa$="i"orab$="I"thenn=1:elseifa$="n"orab$="N"thenn=0:
else210
220 graphic1,1:char1,0,0,str$(fi)+" fok":g=0:h=10:t=g:s=g
230 a=5*sin(fi*π/180):draw1,200,0to200,130to320,130:draw1,195,5to
200,0to205,5
240 draw1,315,125to320,130to315,135:gosub330
250 char1,14,0,s$:char1,38,15,"t":char1,26,1,"s":box1,h,20,50,36,
,1
260 getkeya$:ifa$<>"s"anda$<>"S"then260:elsechar1,14,0,sp$
270 gosub330:draw1,200+t*5,130-s/2:char1,2,3,str$(t),1:t=t+1:s=a*
t*t
280 j=n:gosub340:ifs<250then270:elsechar1,14,4,g$
290 char1,14,2,t$:char1,14,0,u$:getkeya$
300 ifa$=chr$(140)orab$="t"orab$="T"then50:elseifa$="u"orab$="U"then
220
310 if(a$<>"g"anda$<>"G")org=1then290:elsefori=0to4:char1,14,i,sp
$:next
320 g=1:draw1,200,130:fort=1to120:drawto200+t,130-a*t*t/50:next:g
oto290
330 draw1,250,199to250;fi+270:locates;fi+90:x=rdot(0):y=rdot(1):j
=1
340 drawj,x,ytoh;fitoh;fi+90toh;fi+180:return

```

	osztály	csoport	egyéni
új ismeret	●		
alkalmazás	○		
ellenőrzés			

Betöltési idő: 110 s (32 mp.)

Speciális karakter: nincs

A programmal és a számítógéphez kapcsolt eszközzel lehetővé válik a szabadon eső acélgolyó esési idejének mérése. Az 1. ábrán látható kapcsolási rajzon vázolt kísérleti eszköz az 1. botkormánycsatlakozón keresztül van összekötve a számítógéppel. A két, nyíllal jelölt vezeték a csatlakozó 6 és 8 számú érintkezőjéhez kell kötni. Az A jelű kapcsoló átkapcsolásával az elektromágnes elengedi az acélgolyót, ugyanakkor a kapcsoló másik része nyitja a botkormányhoz kapcsolt két vezeték addig meglévő összeköttetését. A program 760-as sorában ekkor a JOY (1) értéke 0 lesz, így megindul az időmérés. Ha a golyó eléri az alsó, B jelű kapcsolót, és zárja azt, akkor a 780-as sorban a JOY (1) függvény újra 128 értéket vesz fel, és ezzel az időmérés (T% értékének növelése mindaddig, amíg JOY (1) = 0) befejeződik.

A kísérleti eszköz elkészítéséhez részletes útmutatót nem adunk, mert úgy gondoljuk, hogy a kapcsolási rajz alapján a készülék megépíthető. (Tájékoztatásul megadjuk, hogy az elektromágnes egy 8V-os villanycsengő egyik tekercse, az áramforrás egy zsebtelep volt a saját célra megépített eszközben. Az A jelzésű kapcsoló egy porszívó kapcsolója volt, míg a B jelzésű egy olyan villanykapcsoló, mely nagymértékű (8\*8 cm-es) billenő része miatt tenyérrel is kapcsolható.) A számítógéphez való csatlakozás előtt mindenképpen olvassuk el a könyv elején lévő ehhez kapcsolódó fejezetet is.

A program indítás után felrajzolja a kísérlet kapcsolási rajzát, és demonstrálja a mérést. A demonstráció az S gomb megnyomásával (Start), illetve bizonyos idő után "magától" is elindul. A demonstráció után vagy megismételtetjük azt, vagy az M gomb megnyomásával a tényleges mérésre térünk át.

A mérés elején a program bekéri az esés magasságát. A beírt adatot a RETURN billentyű megnyomásával le kell zárni. Ezután, ha a kapcsolók nem a megfelelő helyzetben vannak, figyelmeztet erre a tényre, majd megkérdezi, hogy mindkettőt beállítottuk-e?

Itt az I vagy az N gomb egyikének lenyomásával lehet válaszolni. Ezekután kéri, hogy a golyót helyezzük az elektromágneshez, és a kapcsolót (A) kapcsoljuk "start" helyzetbe.

A kapcsolás hatására a gép elvégzi a mérést, majd kiírja a megtett utat, a mért és az útból számított esési időket, valamint a mért idő és a megadott út alapján a nehézségi gyorsulás értékét. Ezután a megjelenő menünek megfelelően megismételhetjük a mérést ugyanezen az úton, vagy az M gomb megnyomásával másik mérésre térhetünk át. A D gombbal a demonstráció ismételt meg.

A mérés hitelességének bizonyításához érdemes először a golyót indítás után megfogni, és csak késve elengedni. Így a tanulók is ellenőrizhetik, hogy a gép és az eszköz valóban mér, és nem csak szimulálja a szabadesést.

A program a RUN/STOP billentyűvel állítható le. A funkcióbillentyűk eredeti állapota a RESET gomb megnyomásával állítható vissza, de ezzel a programot is töröljük a gépből.

```
30 graphic0,1:fori=1to7:keyi,"":next:key8,chr$(8)
```

```
40 color0,2,7:color4,2,7:color1,1
```

```
50 printchr$(142);chr$(8);
```

```
60 b$=" N --N
```

```
70 print" 0-----)|
```

```
80 print" | )| szabadeses merese
```

```
90 print" | )|
```

```
100 print" | Q 0-----P
```

```
110 print" | |U-----I|
```

```
120 print" | | | |
```

```
130 print" | | | |
```

```
140 print" | | start=s | |
```

```
150 print" | | | |
```

```
160 print" | | | |
```

```
170 print" | | J-----K|
```

```
180 print" | | L----->
```

```
190 print" | | | |
```

```
200 print" | | 0-----P
```

```
210 print" | | commodore16 | |
```

```
220 print" | | ~~~~~~ | |
```

```
230 print" | | ~~~~~~ | |
```

```
240 print" | | ~~~~~~ | |
```

```
250 print" | | N ----->
```

```
260 print" | | | |
```

```
270 print" | | | |
```

```
280 print" | | | |
```

```
290 print" | | | |
```

```
300 print" | | | |
```

```
310 print" | | | |
```

```
320 t=0
```

```
330 t=t+1:getc$
```

```
340 ifc$="s"orc$="S"then360
```

```
350 ift<500then330
```

```
360 char1,30,7," "
```

```
370 char1,6,21,b$:char1,32,21,"NOP"
```

```
380 fori=3to16
```

```
390 char1,15,i," Q":char1,30,5,"t =" +str$(i-3)+" "
```

```
400 fort=0to100:next
```



```

410 next
420 char1,15,18," " :char1,32,21," "
430 char1,30,7,"meres=m":char1,30,9,"demo=d"
440 t=0
450 t=t+1:getc$
460 ifc$="m"orc$="M"then500
470 ifc$="d"orc$="D"orc$=chr$(8)thenrun
480 ift<500then450:elserun
490 :
500 rem meres
510 scnclr
520 printchr$(14);
530 print"Hany meter magasrol esik a test? ";
540 b$=""
550 print" ←";:getkeya$
560 ifa$=chr$(13)andb$>" "then640
570 ifa$=chr$(8)thenrun
580 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):goto620
590 ifa$="."then610
600 ifa$>"9"ora$<"0"then550
610 b$=b$a$
620 printa$;
630 iflen(b$)<4then550
640 s=val(b$)
650 scnclr
660 print"Helyezze a kapcsolokat 'kesz' allasba!"
670 ifjoy(1)=128thenchar1,0,5,"Mindkettot beallitotta? (i/n)":else670
680 getkeya$
690 ifa$<>"i"anda$<>"I"then670
700 scnclr
710 print"Helyezze a golyot az elektromagneshez!"
720 print" A kapcsolot 'start' helyzetbe kapcsolva indul a meres"
730 t%=0
740 ifjoy(1)=128then740
750 print"meres"
760 ifjoy(1)=0thent%=t%+1:goto760
770 t=t%/48
780 scnclr
790 print"      S Z A B A D E S E S      M E R E S E"
800 print" Megtett ut : ";s;"m"
810 print" Szamitott esesi ido : ";
820 printusing"#.## s";sqr(2*s/9.81)
830 print" Mert esesi ido :";
840 printusing"##.## s";t
850 print" Mert nehezsegi gyorsulas :";
860 printusing"##.## m/s^2";2*s/t/t
870 char1,0,24,"UJRA = U.      MASIK MERES = M      DEMO = D"
880 t=0
890 t=t+1:getc$
900 ifc$=chr$(8)orc$="d"orc$="D"thenrun
910 ifc$="u"orc$="U"then650
920 ifc$="m"orc$="M"then500
930 ift<1000then890
940 run

```

	osztály	csoport	egyéni
új ismeret	○	●	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 85 s (30 mp.)

Speciális karakter: nincs

A programmal a rezgő testek mozgása szimulálható, és alkalmazása hozzájárulhat a rezgésidő fogalmának kialakításához. Az indítás után megjelenő ismertető elolvasását követően a T gomb megnyomásával mehetünk tovább. Ennek hatására a program felrajzolja az út - idő grafikon koordinátarendszerét, a testet jelképező kis négyzetet és kiírja a START = S üzenetet.

Az S gomb megnyomása előtt a ↑ és a ↓ gombokkal állítható be a kezdeti kitérés. A mozgás elindítása után a test harmonikus rezgőmozgást végez. A rezgőmozgás periódusidejét véletlenszerűen választja meg a program. A mozgással egyidőben az út-idő grafikon néhány pontját is megrajzolja a program, miközben a kitérés pillanatnyi értékét is kiírja a képernyőre. A mozgás befejezése után az alábbi lehetőségek közül választhatunk:

Az U gomb megnyomásával a mozgás újra megismételhető. A T gomb hatására a program továbbmegy egy újabb más rezgésidőjű mozgás vizsgálatához. Az új mozgás út - idő grafikonját az előző koordinátarendszerben ábrázolja a gép. Így a különböző mozgások grafikonjai együtt is tanulmányozhatók. Ha a korábban kapott grafikonokat törölni szeretnénk, nyomjuk meg egyidejűleg a SHIFT és a CLEAR/HOME gombokat.

A G gombbal a koordinátarendszerbe berajzolt pontokhoz a program görbét illeszt (szinuszgörbe). A görbe megrajzolásával egyidőben kiírja a kitérés pillanatnyi értékét. A grafikon megrajzolása után már csak az ÚJRA = U és a TOVABB = T üzenetek jelennek meg.

A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A program leállítása a RUN/STOP gomb megnyomásával történhet. Ha a leállítás grafikus üzemmódban következett be, és ezért a READY üzenet és a villogó kurzor nem jelent meg, a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN billentyű lenyomásával térhetünk vissza a normál üzemmódba. A funkcióbillentyűk átprogramozott állapota a RESET gomb megnyomásával megszüntethető, de ez a program törlésével jár együtt.

```

10 rem rezgomozgas
20 rem zatonyi sandor bekescsaba 1986.
30 deffns(t)=int(100.5-a*cos(w*t)):poke740,212
40 sp$="          ":s$="START=S":t$="TOVABB=T":u$="UJRA=U":g$="GR
AFIKON=G"
50 color0,9,3:color4,9,3:color1,2,7
60 graphic0,1:printchr$(14);chr$(8);:key8,chr$(8):fort=1to7:keyt,
"":next
70 char1,15,3,"REZGOMOZGAS":printsp$+"ZZZZI"
80 print"☐☐☐A program bemutatja egy rezgo test moz-"
90 print"gasat es felrajzolja az ut-ido grafikon nehany pontjat."
100 print"☐Ezekhez a G gombokkal gorbe illesztheto."
110 print"☐A mozgás az U gombokkal ujra indithato, a"
120 print"T gombokkal pedig tovabbi mozgás vizsgal- hato."
130 print"☐A kezdeti kiteres a fel- es lefele muta-to nyilakkal a
llithato be."
140 char1,24,24,t$
150 geta$:ifa$<>"t"anda$<>"T"then150
160 graphic1,1
170 h=180:fi=0:locatel140,280:gosub380:fi=270:gosub380
180 h=7.1:fi=45:locatel135,5:gosub380:locate314,95:gosub380
190 fort=20to180step20:draw1,137,tto140,t:draw1,120+t,100to120+t,
103:next
200 char1,18,1,"x(cm)":char1,35,11,"t(s)"
210 char1,29,14,"5":char1,13,22,"-40":char1,15,12,"0":char1,14,2,
"40"
220 w=int(rnd(0)*10+5)/100:a=20:g=0
230 gosub430
240 t=0:s=100-a:n=1:gosub390:gosub430:n=0:char1,0,0,s$:geta$:ifa$
=chr$(8)thenrun
250 ifa$="s"ora$="S"thenchar1,0,0,sp$:goto280
260 ifa$="☐"anda<80thena=a+2:elseifa$="☐"anda>-80thena=a-2:else240
270 gosub390:g=0:goto240
280 dountilt>180:n=0:gosub390:t=t+4:s=fns(t):n=1:gosub390:draw1,1
40+t,s:gosub430
290 loop:ifg=0thenchar1,0,15,g$
300 char1,0,8,"T =" +str$(int(π/w)/10)+" s"
310 char1,0,17,u$:char1,0,19,t$
320 getkeya$:ifa$=chr$(8)thenrun:elseif(a$="g"ora$="G")andg=0then
350
330 ifa$="☐"then160:elseifa$="u"ora$="U"thengosub400:goto230
340 ifa$="t"ora$="T"thengosub400:goto220:else320
350 t=0:s=fns(t):gosub410:draw,140,180:dountilt=180:drawto140+t,s
:gosub430
360 char1,0,19,"TOVABB=G":geta$:ifa$="g"ora$="G"thent=t+2:s=fns(t
):loop:else360
370 g=1:goto300
380 drawtoh;fitch;fi+90:fi=fi+90:return
390 boxn,90,s,100,s-10:return
400 fort=0to24:char1,0,t,sp$+"          ":next:return
410 forc=9to24:char1,0,c,sp$:next
430 char1,0,6,"x =" +str$(int((100-s)/2))+" cm ":return

```

	osztály	csoport	egyéni
új ismeret	●	○	●
alkalmazás			
ellenőrzés			

Betöltési idő: 110 s (32 mp.)

Speciális karakter: @ → ▲ £ → ▼

[ → ~ ' → ≈

A programmal egyszerű módon vezérelhető a számítógép két beépített hanggenerátora. Az indítás után egy rövid kezelési leírás jelenik meg a képernyőn, melynek elolvasása után a T gomb lenyomásával léphetünk tovább.

A képernyőn két téglalap alakú mező jelenik meg, ezek jelképezik a két generátort. Mindkettőn feltünteti a program a frekvenciájuk pillanatnyi értékét és azt, hogy ki- vagy bekapcsolt állapotban vannak. A baloldali generátoron látható háromszög alakú jel azt mutatja, hogy éppen melyik generátor frekvenciája változtatható a későbbiekben ismertetésre kerülő módon. Ez a jelzés a → illetve a ← gombokkal a megfelelő helyre vihető, és így mindkét generátor hangolható.

A kiválasztott generátort két módon hangolhatjuk: a ↑ és a ↓ gombokkal folyamatosan növelhetjük, illetve csökkenthetjük a frekvenciát. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a generátor frekvenciája törlődik, és a billentyűzet segítségével új frekvencia írható be. A beírást a RETURN billentyű lenyomásával kell lezárni. (Ha a beírt érték 5 karakter, a lezárás automatikusan megtörténik.) A gépen - műszaki okok miatt - a beírt értéknek megfelelő hang nem minden esetben szólaltatható meg, ilyenkor a program a beírthoz legközelebbi lehetséges frekvenciát választja ki. Az elérhető frekvencia 109,3 Hz és 56 kHz között van.

A két hanggenerátor hangja az 1 illetve 2 gombok lenyomásával külön-külön, a 3 gomb megnyomásával együtt szólaltatható meg. A 0 gomb megnyomása mindkét generátort kikapcsolja.

A CLEAR/HOME billentyű megnyomása szintén mindkét generátort kikapcsolja, és mindkettőn 440 Hz frekvenciát állít be (alaphelyzet). A HELP billentyű megnyomása itt is a bevezetőhöz történő visszatérést eredményezi.

A program a RUN/STOP billentyű megnyomásával állítható le. A leállítás után a gép továbbra is a módosított karakterkészletet használja, ezért egy esetleges hiba miatt a képernyő olvashatatlaná válik. Ez az állapot az F1 gomb megnyomásával megszüntethető. Az eredeti karakterkészlet illetve a funkcióbillentyűk eredeti beállítása a RESET gomb megnyomásával állítható vissza, de ügyeljünk arra, hogy ez a program törlésével jár együtt.

```

10 rem hanggenerator
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr:restore
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 fori=0to7:reada:poke15584+i,a:next
120 fori=0to7:reada:poke15576+i,a:next
130 fori=0to7:reada:poke15672+i,a:next
140 poke65298,56:poke65299,56
150 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
160 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
170 data 1,3,7,15,31,63,127,255
180 data 255,127,63,31,15,7,3,1
190 data 0,0,96,144,9,6,0,0
200 data 96,144,9,102,144,9,6,0
210 key1,"poke65298,56:poke65299,56"+chr$(13)
220 fori=2to7:keyi,"":next:key8,chr$(140)
230 color0,3,2:color4,3,2:color1,8
240 printchr$(14);chr$(8);
250 print"  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ "
260 print"  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _  HANGGENERATOR"
270 print"  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ "
280 print" _ _ _ A program segítségével a számítógépbe"
290 print"épített két hanggenerator vezérelhető."
300 print" _ _ _ A két generator közül a jobbra illetve"
310 print"a balra mutató nyíllal lehet választani."
320 print" _ _ _ A felfele illetve lefele mutató nyilak"
330 print"segítségével a rezgésszám változtatható."
340 print" _ _ _ A generátorok hangját az 1-3 gombokkal"
350 print"lehet bekapcsolni, és a 0 gombbal lehet"
360 print"kikapcsolni."
370 print" _ _ _ A SHIFT és CLEAR/HOME együttes lenyomása"
380 print"után új frekvencia írható be."
390 print" _ _ _ A CLEAR/HOME gomb 440 Hz-re állítja be"
400 print"a frekvenciákat."
410 char1,25,24,"TOVABB = _ T _"
420 getkeya$:ifa$=chr$(140)thenrun

```

```

430 ifa$<>"t"anda$<>"T"then420
440 color0,10,2:color4,10,2
450 scnclr:vol7
460 printchr$(14);chr$(8)
470 print"███          HANGGENERATOR"
480 fori=10to18
490 char1,6,i,"█          █          █          █"
500 next
510 k(1)=0:k(2)=0
520 f0=111840.45:f(1)=770:f(2)=770:j=1
530 sound1,0,0:sound2,0,0
540 ifk(1)=1thensound1,f(1),65535
550 ifk(2)=1thensound2,f(2),65535
560 char1,7,15,"█":printusing"#####. # Hz";f0/(1024-f(1))
570 char1,23,15,"█":printusing"#####. # Hz";f0/(1024-f(2))
580 ifk(1)=1thenchar1,11,11,"█BE":elsechar1,11,11,"█KI"
590 ifk(2)=1thenchar1,27,11,"█BE":elsechar1,27,11,"█KI"
600 ifj=1thenchar1,11,17,"█@█":char1,27,17,"█  "
610 ifj=2thenchar1,11,17,"█  ":char1,27,17,"█@█"
620 geta$
630 ifa$="█"thenj=1:goto600
640 ifa$="█"thenj=2:goto600
650 ifa$="0"thenk(1)=0:k(2)=0:goto530
660 ifa$="1"thenk(1)=1:k(2)=0:goto530
670 ifa$="2"thenk(1)=0:k(2)=1:goto530
680 ifa$="3"thenk(1)=1:k(2)=1:goto530
690 ifa$="█"andf(j)<1022thenf(j)=f(j)+1:goto530
700 ifa$="█"andf(j)>1thenf(j)=f(j)-1:goto530
710 ifa$="█"then510
720 ifa$="█"then750
730 ifa$=chr$(140)thenprint:sound1,0,0:sound2,0,0:run
740 goto620
750 b$=""
760 char1,16*j-9,15,"█          "
770 char1,16*j-9,15,"█"
780 getkeya$
790 ifa$=chr$(13)then830
800 ifa$=chr$(20)andb$>" "thenprint"█ █";:b$=left$(b$,len(b$)-1):
goto780
810 ifa$>="0"anda$<="9"thenprinta$;:b$=b$+a$:iflen(b$)>4then830
820 goto780
830 ifval(b$)=0then850
840 f(j)=int(1024.5-f0/val(b$))
850 iff(j)<1thenf(j)=1
860 if f(j)>1022thenf(j)=1022
870 goto530

```

	osztály	csoport	egyéni
új ismeret	○	●	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 85 s (30 mp.)

Speciális karakter: nincs

A program a kölcsönhatások, a lendület, a lendületmegmaradás, illetve az energiamegmaradás törvényének tanításához használható. A programmal egy álló és egy mozgó test rugalmas ütközése szimulálható. A mozgó test tömegét  $m_1$ , sebességét  $v_1$ , az álló test tömegét  $m_2$  jelöli. A két test ütközés utáni sebessége  $u_1$  illetve  $u_2$ .

Az indítás után rövid ismertető olvasható a képernyőn. Innen a T gomb lenyomásával léphetünk tovább. A program ekkor sorban rákérdez az  $m_1$ ,  $m_2$  és a  $v_1$  értékére. A tömegek maximális értéke 10000 kg lehet, a mozgó test sebessége pedig nem haladhatja meg a 75 m/s értékét. Valamennyi adat beírását a RETURN lenyomásával kell lezárni.

A program ezek után a beírt adatoknak megfelelően szimulálja a két test ütközését, majd kiírja az ütközés utáni sebességeket. A tömeg- és sebességadatokból az ütközés előtti és utáni állapotban is meghatározható a két testből álló rendszer lendülete és energiája, valamint az egyes testek lendület-, - illetve energia-változása. Mivel a két test ilyenkor is látszik a képernyőn, lehetőség van az ütközés után eltelt (azonos) idő alatt megtett útjuk lemérésére. (Az ütközés helyét nyíl jelzi.) Az utak arányából szintén következtetni lehet az ütközés utáni sebességek arányára.

A T gomb lenyomására a program lehetőséget ad más adatokkal történő ismétlésére is. Ha valamelyik előzőleg használt adatot nem akarjuk módosítani, a RETURN billentyű lenyomásával továbbléphetünk. Ekkor a program az előzőleg megadott értékekkel dolgozik tovább. Így pl. a RETURN billentyű háromszori megnyomásával az ütközés újra megismételhető az előző paraméterekkel.

A HELP gomb megnyomásával a program elején olvasható bevezető ismertetőhöz lehet visszamenni. A program a RUN/STOP billentyűvel

állítható le. Vigyázzunk arra, hogy a leállítás lehetőleg ne a grafikus üzemmódban történjen, mert ekkor a READY felirat és a villogó kurzor nem jelenik meg a képernyőn. (Ilyen esetben a GRAPHIC 0 "vakon" történő beírásával és a RETURN billentyű lenyomásával térhetünk vissza a normál szöveges üzemmódbhoz.) A funkcióbillentyűk a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával szüntethetjük meg, de ez a programot is törli.

```

10 rem utkozések
20 rem zatonyi sandor bekescsaba 1985.
30 poke740,212:graphic0,1:t$=" test tomege ":v$=" test sebessége
"
40 printchr$(14):printchr$(8):fori=1to7:keyi,"":next:key8,chr$(8)
50 forc=1to3:print"  █  █  █  █":next
60 print"███  █ 1 █——>  █ 2 █  U T K O Z E S E K"
70 print"████ A program testek utkozését szemlelteti."
80 print"█ Az egyik test v1 sebességgel utkozik a másik, eredeti
leg nyugvo testhez."
90 print"█ Valtoztathato a mozgo";v$;" es mindket";t$;". "
100 print"█ A testek tomege maximum 10000 kg, a"
110 print" mozgo";v$;"pedig legfeljebb"
120 print" 75 m/s lehet."
130 char1,25,23,"TOVABB = T":getkeya$:ifa$<"t"then130
140 scnclr:print"███ Az 1.";t$;"(kg) ";:gosub420:ifb$>" "thenm1$=b
$:m1=val(m1$)
150 ifm1=0orm1>10000then140
160 char1,25,2,m1$
170 print:print"███ A 2.";t$;"(kg) ";:gosub420:ifb$>" "thenm2$=b$:
m2=val(m2$)
180 ifm2=0orm2>10000thenprintchr$(27)+"d":gotol160
190 char1,24,5,m2$
200 print:print"███ Az 1.";v$;"(m/s) ";:gosub420:ifb$>" "thenv1$=b
$:v1=val(v1$)
210 ifv1=0orv1>75thenprintchr$(27)+"d":gotol190
220 graphic1,1
230 draw1,0,100to320,100:draw1,160,65to160,80:draw1,155,75to160,8
0to165,75
240 char1,5,1,"m1 = "+m1$+" kg"
250 char1,25,1,"m2 = "+m2$+" kg"
260 char1,5,15,"v1 = "+v1$+" m/s"
270 char1,25,15,"v2 = 0 m/s"
280 u1=(m1-m2)*v1/(m1+m2):u2=2*m1*v1/(m1+m2)
290 c=0:d=160
300 dowhilec+v1<151
310 a=c:b=d:c=a+v1:d=b+0:gosub370:loop
320 a=c:c=151
330 box0,a,90,a+9,99:box1,c,90,c+9,99
340 dowhilec+u1>0andd+u2<311

```



```

350 a=c:b=d:c=a+u1:d=b+u2:gosub370:loop
360 goto380
370 box0,a,90,a+9,99:box1,c,90,c+9,99:box0,b,90,b+9,99:box1,d,90,
d+9,99:return
380 char1,5,18,"u1 =" +str$(.01*int(100*u1))+" m/s"
390 char1,25,18,"u2 =" +str$(.01*int(100*u2))+" m/s"
400 char1,25,23,"TOVABB = T"
410 getkeya$:ifa$="t"thengraphic0:goto140:elseifa$=chr$(8)thenrun
:else410
420 b$=""
430 print"  ←▣▣▣▣";getkeya$:ifa$=chr$(8)thenrun:elseifa$=chr$(13)
then470
440 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1):goto
430
450 if(a$>"9"ora$<"0")anda$<>". "then430:elseprinta$;:b$=b$+a$
460 iflen(b$)<5then430
470 printchr$(27)+"q";:return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 185 s (40 mp.)

Speciális karakter: nincs

A program az  $I = m * v$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább a menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, melynek az értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésköznek 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységet nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezek után három lehetőség közül lehet választani. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket

(állandó, független- illetve függő változó) megváltoztatnánk. Ilyenkor mód van arra is, hogy érték beírása nélkül a RETURN billentyű megnyomásával az előzőleg használt értéket megtartsuk. A program futása a RUN/STOP billentyű megnyomásával szakítható meg. A funkcióbillentyűk azonban a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez egyúttal a program törlését is jelenti.

```

10 rem lendulet (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
60 scnc1r
160 for i=1 to 7: key i, "": next: key 8, chr$(140)
170 f1$="    LENDULET      TOMEG      SEBESSEG"
180 f2$="    (kg*m/s)      (kg)      (m/s)"
190 v1$="LENDULET"
200 v2$="TOMEG"
210 v3$="SEBESSEG"
220 v$="-----"
230 sp$=" "
240 col=11: gosub 1990
250 print chr$(14); chr$(8)
260 print "          I=m*v          LENDULET (tabl.2.)"
270 print "          I=m*v          LENDULET (tabl.2.)"
280 print "          I=m*v          LENDULET (tabl.2.)"
290 print "A program az I = m * v összefüggés elem-
300 print "zesehez készült."
310 print "A három mennyiség közül egyet állandó"
320 print "értékűnek választva egy másik mennyiség"
330 print "értéke szabadon változtatható."
340 print "A program ezek figyelembevételével"
350 print "kiszámítja a harmadik változó megfelelő"
360 print "értékeit."
370 print "A CLEAR/HOME megnyomására új értékek"
380 print "adhatók az egyes mennyiségeknek, a"
390 print "SHIFT és CLEAR/HOME együttes lenyomása"
400 print "után pedig az egyes mennyiségek szerepe"
410 print "változtatható meg."
420 char 1, 25, 24, "TOVABB = "
430 getkey a$: if a$=chr$(140) then run
440 if a$<>"t" and a$<>"T" then 430
450 col=10: gosub 1990
460 all=50: kezd=10: lep=1
470 print "          A VIZSGALT ESET KIVALASZTASA          "+sp$
480 char 1, 7, 5, "FUGGO          FUGGETLEN          "
490 char 1, 18, 6, "ALLANDO"
500 char 1, 6, 7, "VALTOZO          VALTOZO          "
510 char 1, 0, 8, v$
520 char 1, 0, 9, " 1. LENDULET      TOMEG      SEBESSEG      "+sp$
530 char 1, 0, 11, " 2. LENDULET      SEBESSEG      TOMEG      "+v$
540 char 1, 0, 13, " 3. TOMEG      LENDULET      SEBESSEG      "+sp$
550 char 1, 0, 15, " 4. TOMEG      SEBESSEG      LENDULET      "+v$
560 char 1, 0, 17, " 5. SEBESSEG      LENDULET      TOMEG      "+sp$
570 char 1, 0, 19, " 6. SEBESSEG      TOMEG      LENDULET      "+v$

```

```

580 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET VALASZTA
NI."
590 getkey$:ift$=chr$(140)thenrun
600 ift$<"1"ort$>"6"then590
610 col=15:gosub1990
620 print"☐";sp$;" ERTEKADAS " ;sp$
630 char1,2,5,"FUGGO VALTOZO      :"
640 char1,2,7,"Allando mennyiseg  :"
650 char1,2,9,"Fuggetlen valtozo  :"
660 onval(t$)goto670,850,1030,1210,1390,1570
670 :
680 char1,22,5,v1$
690 char1,22,7,v2$
700 char1,22,9,v3$
710 char1,0,11,v$
720 char1,0,14,"A tomeg erteke (kg) : "
730 a=all:p=22:q=14:gosub1750:all=a
740 char1,0,17,"A sebesseg"
750 char1,5,19,"- kezdoerteke (m/s) : "
760 a=kezd:p=27:q=19:gosub1750:kezd=a
770 char1,5,21,"- lepeskoz (m/s) : "
780 a=lep:p=24:q=21:gosub1750:lep=a
790 col=9:gosub1990
800 char1,3,1,"☐—————☐"
810 forj=0to9
820 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
830 next
840 goto1880
850 :
860 char1,22,5,v1$
870 char1,22,7,v3$
880 char1,22,9,v2$
890 char1,0,11,v$
900 char1,0,14,"A sebesseg erteke (m/s) : "
910 a=all:p=26:q=14:gosub1750:all=a
920 char1,0,17,"A tomeg"
930 char1,5,19,"- kezdoerteke (kg) : "
940 a=kezd:p=26:q=19:gosub1750:kezd=a
950 char1,5,21,"- lepeskoz (kg) : "
960 a=lep:p=23:q=21:gosub1750:lep=a
970 col=9:gosub1990
980 char1,3,1,"☐—————☐"
990 forj=0to9
1000 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
1010 next
1020 goto1880
1030 :
1040 char1,22,5,v2$
1050 char1,22,7,v1$
1060 char1,22,9,v3$
1070 char1,0,11,v$
1080 char1,0,14,"A lendulet erteke (kg*m/s) : "
1090 a=all:p=29:q=14:gosub1750:all=a

```

```

1100 char1,0,17,"A sebesseg"
1110 char1,5,19,"- kezdoerteke (m/s) : "
1120 a=kezd:p=27:q=19:gosub1750:kezd=a
1130 char1,5,21,"- lepeskoz (m/s) : "
1140 a=lep:p=24:q=21:gosub1750:lep=a
1150 col=9:gosub1990
1160 char1,17,1," ┌───┐"
1170 forj=0to9
1180 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1190 next
1200 goto1880
1210 :
1220 char1,22,5,v2$
1230 char1,22,7,v3$
1240 char1,22,9,v1$
1250 char1,0,11,v$
1260 char1,0,14,"A sebesseg erteke (m/s) : "
1270 a=all:p=26:q=14:gosub1750:all=a
1280 char1,0,17,"A lendulet"
1290 char1,5,19,"- kezdoerteke (kg*m/s) : "
1300 a=kezd:p=30:q=19:gosub1750:kezd=a
1310 char1,5,21,"- lepeskoz (kg*m/s) : "
1320 a=lep:p=27:q=21:gosub1750:lep=a
1330 col=9:gosub1990
1340 char1,17,1," ┌───┐"
1350 forj=0to9
1360 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1370 next
1380 goto1880
1390 :
1400 char1,22,5,v3$
1410 char1,22,7,v1$
1420 char1,22,9,v2$
1430 char1,0,11,v$
1440 char1,0,14,"A lendulet erteke (kg*m/s) : "
1450 a=all:p=29:q=14:gosub1750:all=a
1460 char1,0,17,"A tomeg"
1470 char1,5,19,"- kezdoerteke (kg) : "
1480 a=kezd:p=26:q=19:gosub1750:kezd=a
1490 char1,5,21,"- lepeskoz (kg) : "
1500 a=lep:p=23:q=21:gosub1750:lep=a
1510 col=9:gosub1990
1520 char1,29,1," ┌───┐"
1530 forj=0to9
1540 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1550 next
1560 goto1880
1570 :
1580 char1,22,5,v3$
1590 char1,22,7,v2$
1600 char1,22,9,v1$
1610 char1,0,11,v$
1620 char1,0,14,"A tomeg erteke (kg) : "

```

```

1630 a=all:p=22:q=14:gosub1750:all=a
1640 char1,0,17,"A lendulet"
1650 char1,5,19,"- kezdoerteke (kg*m/s) : "
1660 a=kezd:p=30:q=19:gosub1750:kezd=a
1670 char1,5,21,"- lepeskoz (kg*m/s) : "
1680 a=lep:p=27:q=21:gosub1750:lep=a
1690 col=9:gosub1990
1700 char1,29,1,"#-----#"
1710 forj=0to9
1720 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1730 next
1740 goto1880
1750 :
1760 b$="":char1,p,q,""
1770 print" ←#####";:getkeya$
1780 ifa$=chr$(13)then1820
1790 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$+a$
1800 ifa$=chr$(20)andb$>""thenprinta$;:b$=left$(b$,len(b$)-1)
1810 iflen(b$)<6then1770
1820 ifb$=""then1850
1830 ifval(b$)<.01thenchar1,p,q,sp$:goto1750
1840 a=val(b$)
1850 char1,p,q,str$(a)+sp$
1860 ifa<1thenchar1,p,q,"0"
1870 return
1880 :
1890 char1,0,0,f1$:char1,0,3,f2$
1900 forj=0to9
1910 char1,0,6+2*j,""
1920 printusing"#####. #";x(j);
1930 printusing"#####. #";y(j);
1940 printusing"#####. #";z(j);
1950 next
1960 getkeya$:ifa$="☒"then610:elseifa$=""then450:elseifa$=chr$(14
0)thenrun
1970 goto1960
1980 stop
1990 :
2000 scnc1r
2010 color0,col,6:color1,col,1:color4,col,6
2020 return

```

## 18. MOZGÁSI ENERGIA (tábl.2.) 8. osztály

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 200 s (41 mp.)

Speciális karakter: @ →  $\pi$

A program az  $E = m \cdot v^2 / 2$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a másik két mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább a menühöz. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, aminek az értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésköznek 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységeket nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezekután három lehetőség közül lehet választani. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühöz térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME billentyű megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket

(állandó, független- illetve függő változó) megváltoztatnánk. Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN billentyű megnyomásával az előzőleg használt értéket megtartsuk. A program futása a RUN/STOP billentyű megnyomásával szakítható meg. A funkcióbillentyűk azonban a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával meg lehet szüntetni, de ez egyúttal a program törlését is jelenti.

```

10 rem mozgasi energia (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next:rem @
110 poke65298,56:poke65299,56
120 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
130 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
140 data112,72,24,32,120,0,0,0
150 key1,"poke65298,56:poke65299,56"+chr$(13)
160 fori=2to7:keyi,"":next:key8,chr$(140)
170 f1$="    ENERGIA          TOMEG          SEBESSEG"
180 f2$="    (J)              (kg)          (m/s)"
190 v1$="ENERGIA"
200 v2$="TOMEG"
210 v3$="SEBESSEG"
220 v$="-----"
230 sp$=" "
240 col=11:gosub1990
250 printchr$(14);chr$(8)
260 print"          MOZGASI ENERGIA"
270 print" ← E=m*v@/2 "
280 print"          (tabl.2.)"
290 print" A program az E = m * v@ / 2 osszefugges"
300 print"elemzesehez keszult."
310 print" A három mennyiség közül egyet allando"
320 print"ertekunek valasztva egy másik mennyiség"
330 print"erteke szabadon változtathato."
340 print" A program ezek figyelembevetelel"
350 print"kiszamítja a harmadik változo megfelelo"
360 print"ertekeit."
370 print" A CLEAR/HOME megnyomasara uj ertekek"
380 print"adhatok az egyes mennyisegeknek, a"
390 print"SHIFT es CLEAR/HOME egyutt es lenyomasa"
400 print"utan pedig az egyes mennyisegek szerepe"
410 print"változtathato meg."
420 char1,25,24,"TOVABB = >T"

```



```

430 getkeya$:ifa$=chr$(140)thenrun
440 ifa$<>"t"anda$<>"T"then430
450 col=10:gosub1990
460 all=50:kezd=10:lep=1
470 print"☐"+sp$+"          A VIZSGALT ESET KIVALASZTASA          "+sp$
480 char1,7,5,"FUGGO          FUGGETLEN          "
490 char1,18,6,"ALLANDO"
500 char1,6,7,"VALTOZO          VALTOZO          "
510 char1,0,8,v$
520 char1,0,9," 1. ENERGIA          TOMEG          SEBESSEG          "+sp$
530 char1,0,11," 2. ENERGIA          SEBESSEG          TOMEG          "+v$
540 char1,0,13," 3. TOMEG          ENERGIA          SEBESSEG          "+sp$
550 char1,0,15," 4. TOMEG          SEBESSEG          ENERGIA          "+v$
560 char1,0,17," 5. SEBESSEG          ENERGIA          TOMEG          "+sp$
570 char1,0,19," 6. SEBESSEG          TOMEG          ENERGIA          "+v$
580 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET          VALASZTA
NI."
590 getkeyt$:ift$=chr$(140)thenrun
600 ift$<"1"ort$>"6"then590
610 col=15:gosub1990
620 print"☐";sp$;"          ERTEKADAS          ";sp$
630 char1,2,5,"FUGGO VALTOZO          : "
640 char1,2,7,"Allando mennyiseg          : "
650 char1,2,9,"Fuggetlen valtozo          : "
660 onval(t$)goto670,850,1030,1210,1390,1570
670 :
680 char1,22,5,v1$
690 char1,22,7,v2$
700 char1,22,9,v3$
710 char1,0,11,v$
720 char1,0,14,"A tomeg erteke (kg)          : "
730 a=all:p=22:q=14:gosub1750:all=a
740 char1,0,17,"A sebesseg"
750 char1,5,19,"- kezdoerteke (m/s)          : "
760 a=kezd:p=27:q=19:gosub1750:kezd=a
770 char1,5,21,"- lepeskoz (m/s)          : "
780 a=lep:p=24:q=21:gosub1750:lep=a
790 col=9:gosub1990
800 char1,3,1,"☐————☐"
810 forj=0to9
820 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)^2/2
830 next
840 goto1880
850 :
860 char1,22,5,v1$
870 char1,22,7,v3$
880 char1,22,9,v2$
890 char1,0,11,v$
900 char1,0,14,"A sebesseg erteke (m/s)          : "
910 a=all:p=26:q=14:gosub1750:all=a
920 char1,0,17,"A tomeg"
930 char1,5,19,"- kezdoerteke (kg)          : "
940 a=kezd:p=26:q=19:gosub1750:kezd=a

```

```

950 char1,5,21,"- lepeskoz (kg) : "
960 a=lep:p=23:q=21:gosub1750:lep=a
970 col=9:gosub1990
980 char1,3,1," ┌──────────┐"
990 forj=0to9
1000 z(j)=all:y(j)=kezd+j*lep:x(j)=all12*y(j)/2
1010 next
1020 gotol880
1030 :
1040 char1,22,5,v2$
1050 char1,22,7,v1$
1060 char1,22,9,v3$
1070 char1,0,11,v$
1080 char1,0,14,"Az energia erteke (J) : "
1090 a=all:p=24:q=14:gosub1750:all=a
1100 char1,0,17,"A sebesseg"
1110 char1,5,19,"- kezdoerteke (m/s) : "
1120 a=kezd:p=27:q=19:gosub1750:kezd=a
1130 char1,5,21,"- lepeskoz (m/s) : "
1140 a=lep:p=24:q=21:gosub1750:lep=a
1150 col=9:gosub1990
1160 char1,17,1," ┌──────────┐"
1170 forj=0to9
1180 x(j)=all:z(j)=kezd+j*lep:y(j)=all*2/z(j)12
1190 next
1200 gotol880
1210 :
1220 char1,22,5,v2$
1230 char1,22,7,v3$
1240 char1,22,9,v1$
1250 char1,0,11,v$
1260 char1,0,14,"A sebesseg erteke (m/s) : "
1270 a=all:p=26:q=14:gosub1750:all=a
1280 char1,0,17,"Az energia"
1290 char1,5,19,"- kezdoerteke (J) : "
1300 a=kezd:p=25:q=19:gosub1750:kezd=a
1310 char1,5,21,"- lepeskoz (J) : "
1320 a=lep:p=22:q=21:gosub1750:lep=a
1330 col=9:gosub1990
1340 char1,17,1," ┌──────────┐"
1350 forj=0to9
1360 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)*2/all12
1370 next
1380 gotol880
1390 :
1400 char1,22,5,v3$
1410 char1,22,7,v1$
1420 char1,22,9,v2$
1430 char1,0,11,v$
1440 char1,0,14,"Az energia erteke (J) : "
1450 a=all:p=24:q=14:gosub1750:all=a
1460 char1,0,17,"A tomeg"
1470 char1,5,19,"- kezdoerteke (kg) : "

```

```

1480 a=kezd:p=26:q=19:gosub1750:kezd=a
1490 char1,5,21,"- lepeskoz (kg) : "
1500 a=lep:p=23:q=21:gosub1750:lep=a
1510 col=9:gosub1990
1520 char1,29,1," : _____:"
1530 forj=0to9
1540 x(j)=all:y(j)=kezd+j*lep:z(j)=sqr(all*2/y(j))
1550 next
1560 goto1880
1580 char1,22,5,v3$
1590 char1,22,7,v2$
1600 char1,22,9,v1$
1610 char1,0,11,v$
1620 char1,0,14,"A tomeg erteke (kg) : "
1630 a=all:p=22:q=14:gosub1750:all=a
1640 char1,0,17,"Az energia"
1650 char1,5,19,"- kezdoerteke (J) : "
1660 a=kezd:p=25:q=19:gosub1750:kezd=a
1670 char1,5,21,"- lepeskoz (J) : "
1680 a=lep:p=22:q=21:gosub1750:lep=a
1690 col=9:gosub1990
1700 char1,29,1," : _____:"
1710 forj=0to9
1720 y(j)=all:x(j)=kezd+j*lep:z(j)=sqr(x(j)*2/all)
1730 next
1740 goto1880
1760 b$="":char1,p,q," "
1770 print" ←";getkeya$
1780 ifa$=chr$(13)then1820
1790 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$+a$
1800 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1810 iflen(b$)<6then1770
1820 ifb$=""then1850
1830 ifval(b$)<0.01thenchar1,p,q,sp$:goto1750
1840 a=val(b$)
1850 char1,p,q,str$(a)+sp$
1860 ifa<1thenchar1,p,q,"0"
1870 return
1890 char1,0,0,f1$:char1,0,3,f2$
1900 forj=0to9
1910 char1,0,6+2*j," "
1920 printusing"#####. #";x(j);
1930 printusing"#####. #";y(j);
1940 printusing"#####. #";z(j);
1950 next
1960 getkeya$:ifa$=" "then610:elseifa$=" "then450:elseifa$=chr$(14)
0)thenrun
1970 goto1960
1980 stop
2000 scnc1r
2010 color0,col,6:color1,col,1:color4,col,6
2020 return

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 195 s (40 mp.)

Speciális karakter: nincs

A program a  $W = F \cdot s$  összefüggés feladatokban történő alkalmazását gyakoroltatja. A program elindítása után egy rövid ismertető jelenik meg. Ennek elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő menü alapján választhatjuk ki, hogy melyik mennyiség kiszámítását akarjuk gyakorolni. A választáshoz csak a megfelelő számot kell lenyomni. Itt nyílik lehetőség arra is, hogy a 9 gomb lenyomásával az addig elért eredményeket kiírassuk a képernyőre. A HELP billentyű megnyomásával a program elején található bevezetőhöz lehet visszalépni.

A megfelelő mennyiség kiválasztása után egy feladat jelenik meg, melynek szövege, illetve adatai véletlenszerűen változnak. A HELP billentyű megnyomására a gép megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be eredménybe.

A választ a billentyűzet segítségével lehet beírni, és a RETURN billentyű megnyomásával kell lezárni. A válaszba a mértékegységet is be kell írni. A mértékegységet rövidítve (J, N, m) vagy teljes nevükkel (Joule, Newton, méter) is beírhatjuk.

A gép a választ elemzi, és hiba esetén annak jellegére utaló üzenet kiírása után megismétli a kérdést. Ha a második válasz is helytelen, a megoldást is kiírja a képernyőre. Helyes válasz esetén csak a HELYES felirat jelenik meg. Ezután az utolsó sorban megjelenő kiírásnak megfelelően az U gombbal új feladat kérhető, vagy az M gombbal a menühöz térhetünk vissza, és más típusú feladatot választhatunk. A HELP billentyűvel ez esetben is a bevezetőhöz lehet visszamenni.

Az elért eredmények kiírását a menü alapján a 9 gomb lenyomásával kérhetjük. A képernyőn típusonként és összesítve is megjelenik a kapott feladatok száma, az elért pontszám és az eredmény (százalékban kifejezve). Egy feladatra 2, illetve 1 pont kapható aszerint, hogy az első vagy csak a második válasz volt a helyes. A T gombbal a menühöz a HELP billentyű segítségével pedig a bevezetőhöz lehet visszajutni.

A program a RUN/STOP billentyűvel állítható le. A program leállítása után a funkcióbillentyűk átprogramozott állapotban maradnak. Ez a helyzet a RESET gomb megnyomásával szüntethető meg, de ez az eljárás a programot is kitörli a memóriából.

```

10 rem munka (gyak.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 fori=1to9
40 readk$(i),t(i),p1(i),p2(i),d1(i),d2(i)
50 next
60 graphic0,1
70 fori=1to7:keyi,"":next:key8,chr$(140)
80 col=10:gosub930
90 printchr$(14);chr$(8)
100 print"
110 print" ← W=F*s MUNKA (gyak.1.)"
120 print"
130 print" A program a  $W = F * s$  összefüggés fel-"
140 print" adatokban történő alkalmazását gyakoroltatja."
150 print" A válaszokba a meritekegysegeket is be kell írni."
160 print" A választ a RETURN gomb megnyomásával kell lezárni."
170 print" A program jegyzi az elért eredményt."
180 char1,25,24,"TOVABB = T"
190 getkeya$:ifa$="t"ora$="T"then200:elseifa$=chr$(140)then60:else
e190
200 scnc1r
210 col=7:gosub930
220 char1,5,3,"1. Munka |"
230 char1,20,4," |"
240 char1,5,5,"2. Elmozdulas > kiszamitasa"
250 char1,20,6," |"
260 char1,5,7,"3. Ero |"
270 char1,5,9,"4. Vegyes feladatok"
280 char1,5,13,"5. Munka |"
290 char1,20,14," | kiszamitasa"
300 char1,5,15,"6. Elmozdulas >"
310 char1,20,16," | szoveg alapjan"
320 char1,5,17,"7. Ero |"
330 char1,5,19,"8. Vegyes szoveges feladatok"
340 char1,5,23,"9. EREDMENYEK"
350 getkeyt$:ift$=chr$(140)then60
360 t=val(t$)
370 ift<1ort>9then350
380 ift=9then1610
390 ift=1theni=1+3*rnd(0)
400 ift=2theni=4+3*rnd(0)
410 ift=3theni=7+3*rnd(0)
420 ift=4thent=int(1+3*rnd(0)):goto390
430 ift=5theni=1+3*rnd(0):t=4
440 ift=6theni=4+3*rnd(0):t=5
450 ift=7theni=7+3*rnd(0):t=6
460 ift=8thent=int(5+3*rnd(0)):goto430
470 :
480 k$=k$(i):p1=p1(i):p2=p2(i):d1=d1(i):d2=d2(i)

```

```

490 scncclr:col=11:gosub930
500 x=int(34+66*rnd(0))*10^(d1-2)
510 y=int(8*rnd(0)+2)*10^(d2-1)
520 z=x*y
530 x$=str$(x):ifx<1thenx$=" 0"+right$(x$,len(x$)-1)
540 y$=str$(y):ify<1theny$=" 0"+right$(y$,len(y$)-1)
550 z$=str$(z):ifz<1thenz$=" 0"+right$(z$,len(z$)-1)
560 r=0
570 ontgosub970,1020,1070,1120,1180,1240
580 char1,7,7,"←¶"
590 a$="":gosub1320:ifa$=chr$(140)thenprint" ":goto790
600 h=0
610 ifabs(val(b$)-e)>0.0001thenh=1
620 n=1000000
630 ifabs(val(b$)-e*n)<0.0001andn<>1thenh=2
640 n=n/10:ifn>.0000001then630
650 lb=len(b$):l1=len(m1$):l2=len(m2$)
660 ifright$(b$,l1)<>m1$then680
670 iflb>l1+1thenifval(left$(b$,lb-l1-1))=val(b$)then700:else710
680 ifright$(b$,l2)<>m2$then700
690 iflb>l2+1thenifval(left$(b$,lb-l2-1))=val(b$)then700:else710
700 h=h+4
710 char1,0,10,""
720 if(hand1)=1thenprint"          Szamitasi hiba¶"
730 if(hand2)=2thenprint"          Nagysagrendi hiba¶"
740 if(hand4)=4thenprint"          Mertekegyseg hiba"
750 ifh=0then880
760 r=r+1:r(t)=r(t)+1
770 tt=2000:gosub1410
780 ifr=1thenscncclr:goto570
790 char1,0,10,chr$(27)+"q"
800 char1,0,12,chr$(27)+"q"
810 ontgosub1430,1490,1550,1430,1490,1550
820 char1,7,24,"UJ FELADAT =  ¶  MENU =  ¶"
830 getkeya$:ifa$=chr$(140)then60
840 ifa$="u"ora$="U"then360
850 ifa$="m"ora$="M"then200
860 goto830
870 :
880 j(t)=j(t)+2-r
890 char1,7,14,"Helyes. "
900 tt=1000:gosub1410
910 goto820
920 :
930 :
940 color0,col,1:color1,col,7:color4,col,1
950 return
960 :
970 char1,1,1,"F ="+x$+" N"
980 char1,1,3,"s ="+y$+" m"
990 char1,1,5,"_____ "
1000 char1,1,7,"W = ? ":gotol160
1010 return

```

```

1020 char1,1,1,"W =" +z$+" J"
1030 char1,1,3,"F =" +x$+" m"
1040 char1,1,5,"_____ "
1050 char1,1,7,"s = ?  ":gotol220
1060 return
1070 char1,1,1,"W =" +z$+" J"
1080 char1,1,3,"s =" +y$+" m"
1090 char1,1,5,"_____ "
1100 char1,1,7,"F = ?  ":gotol280
1110 return
1120 char1,0,1,k$
1130 char1,0,5,"Mekkora munkat vegzett?"
1140 char1,p1,1,x$
1150 char1,p2,1,y$
1160 e=z:m1$="J":m2$="joule"
1170 return
1180 char1,0,1,k$
1190 char1,0,5,"Mekkora volt az elmozdulas?"
1200 char1,p1,1,z$
1210 char1,p2,1,x$
1220 e=y:m1$="m":m2$="meter"
1230 return
1240 char1,0,1,k$
1250 char1,0,5,"Mekkora erot fejtett ki?"
1260 char1,p1,1,z$
1270 char1,p2,1,y$
1280 e=x:m1$="N":m2$="newton"
1290 return
1300 :
1310 :
1320 b$=""
1330 getkeya$:ifa$=chr$(140)thenreturn
1340 ifa$=chr$(13)andb$<>"thenprint" ":return
1350 ifa$=chr$(20)andb$>"thenprinta$;"← ███";:b$=left$(b$,len(b$)
-1):gotol330
1360 if(a$<="9"anda$>="0")or(a$<="z"anda$>="a")or(a$<="Z"anda$>="
A")then1390
1370 ifa$="."ora$="*"ora$="/"ora$=" "then1390
1380 gotol330
1390 printa$;"← ███";:b$=b$+a$:iflen(b$)>15thenprint" ":return
1400 gotol330
1410 :
1420 fortt=ttto0step-1:next:return
1430 :
1440 char1,7,12,"W = F * s"
1450 char1,7,14,"W =" +x$+" N *"+y$+" m"
1460 char1,7,16,"W =" +z$+" N*m"
1470 char1,7,18,"W =" +z$+" J"
1480 return
1490 :
1500 char1,7,12,"s = W / F"
1510 char1,7,14,"s =" +z$+" J /"+x$+" N"
1520 char1,7,16,"s =" +y$+" J/N"

```

```

1530 char1,7,18,"s =" +y$+" m"
1540 return
1550 :
1560 char1,7,12,"F = W / s"
1570 char1,7,14,"F =" +z$+" J /" +y$+" m"
1580 char1,7,16,"F =" +x$+" J/m"
1590 char1,7,18,"F =" +x$+" N"
1600 return
1610 :
1620 scnclr
1630 col=2:gosub930
1640 char1,17,0,"feladat pont eredmeny"
1650 char1,0,3,"Munka"
1660 char1,0,5,"Elmozdulas"
1670 char1,0,7,"Ero"
1680 char1,0,11,"Munka (szov.)"
1690 char1,0,13,"Elmozdulas (szov.)"
1700 char1,0,15,"Ero (szov.)"
1710 char1,0,19,"OSSZESEN"
1720 fori=1to3
1730 j=j(i):n=j+r(i)
1740 char1,19,1+2*i,"":printusing"###";n/2
1750 char1,26,1+2*i,"":printusing"###";j
1760 ifn=0then1790
1770 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1780 char1,34,1+2*i,"":printusing"### %";j*100/n
1790 color1,2:next
1800 fori=4to6
1810 j=j(i):n=j+r(i)
1820 char1,19,3+2*i,"":printusing"###";n/2
1830 char1,26,3+2*i,"":printusing"###";j
1840 ifn=0then1870
1850 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1860 char1,34,3+2*i,"":printusing"### %";j*100/n
1870 color1,2:next
1880 color1,8,6
1890 j=0:r=0:fori=1to6:j=j+j(i):r=r+r(i):next:n=r+j
1900 char1,19,19,"":printusing"###";n/2
1910 char1,26,19,"":printusing"###";j
1920 ifn=0then1940
1930 char1,34,19,"":printusing"### %";j*100/n
1940 color1,2
1950 goto180
1960 data"Egy ember .. N erovel, ... m uton tolt egy talicskat.
"
1970 data1,9,22,2,3
1980 data"Egy mozdony .... N erovel, ... m uton huzott egy sze
relvenyt."
1990 data1,11,26,4,3
2000 data"Egy daru .... N erovel emelt fel .. m magasra egy te
stet."

```



2010 data1,8,32,4,2	
2020 data"Egy vontato ..... J munkaval, ... N egy potkocsit."	erovel huzott
2030 data2,11,30,3,3	
2040 data"Egy hajo ..... J munkaval, .... N egy uszalyt."	erovel huzott
2050 data2,8,28,4,3	
2060 data"Egy fiu .... J munkaval, ... N erovel nkot."	huzott egy sza
2070 data2,7,24,2,2	
2080 data"Egy daru ..... J munkaval .. m egy testet."	magasra emelt
2090 data3,8,26,4,2	
2100 data"Egy elefant ..... J munkaval .. m-re faronkot."	huzott el egy
2110 data3,11,28,3,2	
2120 data"Egy lo ..... J munkaval .. m tavolra szekeret."	huzott el egy
2130 data3,6,23,3,2	

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 185 s (39 mp.)

Speciális karakter: nincs

A program a munka fogalmának elmélyítéséhez készült. A  $W = F * s$  összefüggésben szereplő három mennyiség közül a gép 2 - 2 mennyiség nagyságrendi viszonyait megadja, és ennek ismeretében kell a tanulónak a hiányzó mennyiségeket összehasonlítani. (Pl. a gép megadja, hogy az erők egyenlők, az egyik test elmozdulása pedig kétszer nagyobb a másik elmozdulásánál. Ebben az esetben a munkákat kell összehasonlítani.)

Elindítás után egy rövid tájékoztatót ad a program. Ezt elolvasva a T gomb lenyomásával lehet továbblépni. Ezután történik a nehézségi fokozat kiválasztása. A felsorolt lehetőségek közül a megfelelő szám lenyomásával lehet választani. Itt van mód arra, hogy az addig elért eredményt a 4 gomb lenyomásával kiírassuk a képernyőre. A HELP billentyű lenyomásával a bevezetőhöz lehet visszamenni.

A nehézségi fokozat meghatározása után az összehasonlítandó mennyiségeket kell kiválasztani. Itt is a megfelelő szám lenyomásával választhatunk. Ezután a program a választásokat figyelembe véve véletlenszerű értéket ad 2 - 2 mennyiségnek, és rákérdez a hiányzó mennyiségek összehasonlításának eredményére. A választ a < = > és a ? gombok segítségével lehet beírni. A ? gombot akkor kell lenyomni, ha a kérdésre nem adható egyértelmű válasz. A HELP billentyű megnyomására a program megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az eredménybe. A válasz beírása után, ha az hibás volt, a program az erre utaló üzenet kiírása után megadja a helyes választ, ellenkező esetben csak a REMEK felirat jelenik meg a képernyőn. Ezután az U gomb megnyomásával új feladatot kérhetünk, az M gomb megnyomásával pedig az első menühez (nehézségi fokozat kiválasztása és eredmények) mehetünk vissza. A HELP billentyű megnyomása ebben az esetben a bevezetőhöz való visszatérést eredményezi.

Az eredmények kiírását az említett módon a menü alapján a 4 gomb lenyomásával kérhetjük. A program nehézségi fokozat és az össze-

hasonlítandó mennyiség szerint csoportosítva, valamint összesítve százalékban adja meg az elért eredményt. Az eredmények kiírása után a T gombbal újra a menühöz lehet visszajutni, míg a HELP billentyű lenyomásával a bevezető ismertetéshez léphetünk vissza. A program a RUN/STOP billentyűvel állítható le. A leállítás után a funkcióbillentyűk átállított állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez a program törlését eredményezi.

```

10 rem munka (gyak 2.)
20 rem zatonyi sandor bekescsaba 1986.
30 scnclr
40 fori=1to7:keyi,"":next:key8,chr$(140)
50 fok$(1)="Konnyu ":t$(1)="Munka"
60 fok$(2)="Kozepes":t$(2)="Ero"
70 fok$(3)="Nehez ":t$(3)="Elmozdulas"
80 t$(4)="VEGYES"
90 col=11:gosub2020
100 printchr$(14);chr$(8)
110 print"
120 print" ← W=F*s MUNKA (gyak.2)"
130 print" _____"
140 print" A program a W = F * s osszefugges"
150 print" elemzesehez keszult."
160 print" A megadott adatok alapjan ossze kell"
170 print" hasonlitani a ket mennyiseget."
180 print" Az osszehasonlitas eredmenyet a < = >"
190 print" gombokkal lehet beirni."
200 print" Ha nem adhato egyertelmu valasz, akkor"
210 print" a ? gombot kell megnyomni."
220 char1,25,24,"TOVABB = T"
230 getkeya$:ifa$=chr$(140)then30
240 ifa$<>"t"anda$<>"T"then230
250 scnclr
260 col=10:gosub2020
270 char1,5,0,"NEHEZSEGI FOKOZAT VALASZTAS"
280 char1,14,7,"1. "+fok$(1)
290 char1,14,10,"2. "+fok$(2)
300 char1,14,13,"3. "+fok$(3)
310 char1,14,18,"4. EREDMENYEK"
320 char1,2,24,"A szam lenyomasaval lehet valasztani."
330 getkeya$:ifa$=chr$(140)then30
340 ifa$="4"then1740
350 ifa$<"1"ora$>"3"then330
360 f=val(a$)
370 scnclr
380 rem col=14:gosub2020
390 gosub2050
400 char1,4,7,"AZ OSSZEHASONLITANDO MENNYISEG"
410 char1,16,12,"1. "+t$(1)
420 char1,16,14,"2. "+t$(2)
430 char1,16,16,"3. "+t$(3)
440 char1,16,20,"4. "+t$(4)

```

```

450 char1,2,24,"A szam lenyomasaval lehet valasztani."
460 getkeyt$:ift$=chr$(140)then30
470 ift$<"1"ort$>"4"then460
480 t=val(t$):ift=4thent=1+3*rnd(0)
490 scncly
500 col=11:gosub2020
510 gosub2050
520 x1=int(1+9*rnd(0)):x2=int(1+9*rnd(0))
530 y1=int(1+9*rnd(0)):y2=int(1+9*rnd(0))
540 r=rnd(0)
550 ifr<.3then580
560 ifr<.65thenx1=x2
570 ifr>=.65theny1=y2
580 ontgosub810,1120,1430
590 char1,8,5,x1$:char1,27,5,x2$
600 char1,8,9,y1$:char1,27,9,y2$
610 char1,7,11,"-----"
620 char1,8,13,z1$:char1,27,13,z2$
630 char1,19,13,"  ?  "
640 getkeya$:ifa$=chr$(140)then740
650 ifa$=","ora$="."ora$="/"thena$=chr$(asc(a$)+16)
660 ifa$<>"<"anda$<>"="anda$<>">"anda$<>"?"then640:elsechar1,19,1
3,a$
670 n(f,t)=n(f,t)+1
680 ifa$=e$then690:else720
690 j(f,t)=j(f,t)+1
700 char1,17,18,"REMEK"
710 goto760
720 char1,17,16,"ROSSZ"
730 fort=0to300:next
740 char1,11,20,"A helyes valasz:  " + e$ + " "
750 fort=0to900:next
760 char1,5,24,"UJ FELADAT = U          MENU = M"
770 getkeya$:ifa$=chr$(140)then30
780 ifa$="m"ora$="M"then250
790 ifa$="u"ora$="U"then480
800 goto770
810 :
820 z1$=" W1":z2$=" W2"
830 onfgoto840,890,1000
840 :
850 x1$=str$(x1)+"N":x2$=str$(x2)+"N"
860 y1$=str$(y1)+"m":y2$=str$(y2)+"m"
870 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
"
880 return
890 :
900 x1$=str$(x1)+"F":x2$=str$(x2)+"F"
910 ifx1=1thenx1$=" F"
920 ifx2=1thenx2$=" F"
930 ifx1=x2thenx1$=" F":x2$=x1$
940 y1$=str$(y1)+"s":y2$=str$(y2)+"s"
950 ify1=1theny1$=" s"

```

```

960 ify2=1theny2$=" s"
970 ify1=y2theny1$=" s":y2$=y1$
980 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
"
990 return
1000 :
1010 x1$=" F1":x2$=" F2"
1020 y1$=" s1":y2$=" s2"
1030 ifx1=x2andy1=y2thene$="=":gotol070
1040 ifx1<=x2andy1<=y2thene$="<":gotol070
1050 ifx1>=x2andy1>=y2thene$=">":gotol070
1060 e$="?"
1070 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1080 char1,19,5,r$
1090 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1100 char1,19,9,r$
1110 return
1120 :
1130 z1$=" F1":z2$=" F2"
1140 onfgotol160,1200,1310
1150 :
1160 x1$=str$(x1)+"J":x2$=str$(x2)+"J"
1170 y1$=str$(y1)+"m":y2$=str$(y2)+"m"
1180 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$=">"
"
1190 return
1200 :
1210 x1$=str$(x1)+"W":x2$=str$(x2)+"W"
1220 ifx1=1thenx1$=" W"
1230 ifx2=1thenx2$=" W"
1240 ifx1=x2thenx1$=" W":x2$=x1$
1250 y1$=str$(y1)+"s":y2$=str$(y2)+"s"
1260 ify1=1theny1$=" s"
1270 ify2=1theny2$=" s"
1280 ify1=y2theny1$=" s":y2$=y1$
1290 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$=">"
"
1300 return
1310 :
1320 x1$=" W1":x2$=" W2"
1330 y1$=" s1":y2$=" s2"
1340 ifx1=x2andy1=y2thene$="=":gotol380
1350 ifx1<=x2andy1>=y2thene$="<":gotol380
1360 ifx1>=x2andy1<=y2thene$=">":gotol380
1370 e$="?"
1380 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1390 char1,19,5,r$
1400 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1410 char1,19,9,r$
1420 return
1430 :

```

```

1440 z1$=" s:1":z2$=" s:2"
1450 onfgoto1460,1510,1620
1460 :
1470 x1$=str$(x1)+"J":x2$=str$(x2)+"J"
1480 y1$=str$(y1)+"N":y2$=str$(y2)+"N"
1490 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1500 return
1510 :
1520 x1$=str$(x1)+"W":x2$=str$(x2)+"W"
1530 ifx1=1thenx1$=" W"
1540 ifx2=1thenx2$=" W"
1550 ifx1=x2thenx1$=" W":x2$=x1$
1560 y1$=str$(y1)+"F":y2$=str$(y2)+"F"
1570 ify1=1theny1$=" F"
1580 ify2=1theny2$=" F"
1590 ify1=y2theny1$=" F":y2$=y1$
1600 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1610 return
1620 :
1630 x1$=" W:1":x2$=" W:2"
1640 y1$=" F:1":y2$=" F:2"
1650 ifx1=x2andyl=y2thene$="=":goto1690
1660 ifx1<=x2andyl>=y2thene$="<":goto1690
1670 ifx1>=x2andyl<=y2thene$=">":goto1690
1680 e$="?"
1690 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1700 char1,19,5,r$
1710 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1720 char1,19,9,r$
1730 return
1740 :
1750 col=15:gosub2020
1760 scnc1r
1770 char1,15,0,"EREDMENYEK"
1780 char1,14,5,"1. 2. 3. OSSZ."
1790 char1,0,9,t$(1)
1800 char1,0,12,t$(2)
1810 char1,0,15,t$(3)
1820 char1,0,19,"OSSZESEN"
1830 fort=1to3
1840 n(4,t)=0:j(4,t)=0
1850 forf=1to3
1860 ifn(f,t)>0thene(f,t)=100*j(f,t)/n(f,t)
1870 n(4,t)=n(4,t)+n(f,t)
1880 j(4,t)=j(4,t)+j(f,t)
1890 next
1900 ifn(4,t)>0thene(4,t)=100*j(4,t)/n(4,t)
1910 next
1920 forf=1to4
1930 n(f,4)=0:j(f,4)=0
1940 fort=1to3

```

```

1950 ifn(f,t)>0thenchar1,5+7*f,6+3*t,"":printusing"###%";e(f,t)
1960 n(f,4)=n(f,4)+n(f,t)
1970 j(f,4)=j(f,4)+j(f,t)
1980 next
1990 ifn(f,4)>0thenchar1,5+7*f,19,"":printusing"###%";100*j(f,4)/
n(f,4)
2000 next
2010 goto220
2020 :
2030 color0,col,2:color1,col,7:color4,col,2
2040 return
2050 :
2060 char1,0,0," "
2070 char1,0,1," Fokozat: "+fok$(f)+" "
2080 char1,0,2," "
2090 return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 185 s (39 mp.)

Speciális karakter: nincs

A program a  $W = F * s$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább a menühöz. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, amelynek értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésköznek 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységeket nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezekután három lehetőség közül választhatunk. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühöz térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.



Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN megnyomásával az előzőleg használt értéket megtartsuk.

A program futása a RUN/STOP billentyű megnyomásával szakítható meg. A funkcióbillentyűk azonban a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával meg lehet szüntetni, de ez egyúttal a program törlését is jelenti.

```
10 rem munka (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 for i=1 to 7: key i, "" : next: key 8, chr$(140)
40 f1$="      MUNKA      ERO      ELMOZDULAS"
50 f2$="      (J)      (N)      (m)"
60 v1$="Munka"
70 v2$="Ero"
80 v3$="Elmozdulas"
90 v$="-----"
100 sp$=" "
110 col=11: gosub 1860
120 print chr$(14); chr$(8)
130 print "      "
140 print "      ← W=F*s      MUNKA (tabl.2)"
150 print "      "
160 print "A program a W = F * s összefüggés"
170 print "elemzéséhez készült."
180 print "A három mennyiség közül egyet állandó"
190 print "értékűnek választva egy másik mennyiség"
200 print "értéke szabadon változtatható."
210 print "A program ezek figyelembevételével"
220 print "kiszámítja a harmadik változó megfelelő"
230 print "értékeit."
240 print "A CLEAR/HOME megnyomására új értékek"
250 print "adhatók az egyes mennyiségeknek, a"
260 print "SHIFT és CLEAR/HOME együttes lenyomása"
270 print "után pedig az egyes mennyiségek szerepe"
280 print "változtatható meg."
290 char 1, 25, 24, "TOVABB = T"
300 getkey a$: if a$=chr$(140) then run
310 if a$<>"t" and a$<>"T" then 300
320 col=10: gosub 1860
330 all=100: kezd=10: lep=1
340 print " " + sp$ + "      A VIZSGALT ESET KIVALASZTASA      " + sp$
350 char 1, 7, 5, "FUGGO      FUGGETLEN      "
360 char 1, 18, 6, "ALLANDO"
370 char 1, 6, 7, "VALTOZO      VALTOZO      "
380 char 1, 0, 8, v$
390 char 1, 0, 9, " 1. Munka      Ero      Elmozdulas      " + sp$
400 char 1, 0, 11, " 2. Munka      Elmozdulas      Ero      " + v$
410 char 1, 0, 13, " 3. Ero      Munka      Elmozdulas      " + sp$
420 char 1, 0, 15, " 4. Ero      Elmozdulas      Munka      " + v$
430 char 1, 0, 17, " 5. Elmozdulas      Munka      Ero      " + sp$
440 char 1, 0, 19, " 6. Elmozdulas      Ero      Munka      " + v$
450 char 1, 0, 23, " A MEGFELELO SZAM LENYOMASAVAL LEHET      VALASZTA
NI. "
```

```

460 getkey$t$:ift$=chr$(140)thenrun
470 ift$<"1"ort$>"6"then460
480 col=15:gosub1860
490 print" ";sp$;" ERTEKADAS " ;sp$
500 char1,5,5,"FUGGO VALTOZO ":"
510 char1,5,7,"Allando mennyiseg ":"
520 char1,5,9,"Fuggetlen valtozo ":"
530 onval(t$)goto540,720,900,1080,1260,1440
540 :
550 char1,25,5,v1$
560 char1,25,7,v2$
570 char1,25,9,v3$
580 char1,0,11,v$
590 char1,0,14,"Az ero erteke (N) : "
600 a=all:p=20:q=14:gosub1620:all=a
610 char1,0,17,"Az elmozdulas"
620 char1,5,19,"- kezdoerteke (m) : "
630 a=kezd:p=25:q=19:gosub1620:kezd=a
640 char1,5,21,"- lepeskoz (m) : "
650 a=lep:p=22:q=21:gosub1620:lep=a
660 col=9:gosub1860
670 char1,4,1," ┌───┐"
680 forj=0to9
690 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
700 next
710 goto1750
720 :
730 char1,25,5,v1$
740 char1,25,7,v3$
750 char1,25,9,v2$
760 char1,0,11,v$
770 char1,0,14,"Az elmozdulas erteke (m) : "
780 a=all:p=27:q=14:gosub1620:all=a
790 char1,0,17,"Az ero"
800 char1,5,19,"- kezdoerteke (N) : "
810 a=kezd:p=25:q=19:gosub1620:kezd=a
820 char1,5,21,"- lepeskoz (N) : "
830 a=lep:p=22:q=21:gosub1620:lep=a
840 col=9:gosub1860
850 char1,4,1," ┌───┐"
860 forj=0to9
870 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
880 next
890 goto1750
900 :
910 char1,25,5,v2$
920 char1,25,7,v1$
930 char1,25,9,v3$
940 char1,0,11,v$
950 char1,0,14,"A munka erteke (J) : "
960 a=all:p=21:q=14:gosub1620:all=a
970 char1,0,17,"A elmozdulas"

```

```

980 char1,5,19,"- kezdoerteke (m) : "
990 a=kezd:p=25:q=19:gosub1620:kezd=a
1000 char1,5,21,"- lepeskoz (m) : "
1010 a=lep:p=22:q=21:gosub1620:lep=a
1020 col=9:gosub1860
1030 char1,18,1,"  ────▶"
1040 forj=0to9
1050 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1060 next
1070 gotol750
1080 :
1090 char1,25,5,v2$
1100 char1,25,7,v3$
1110 char1,25,9,v1$
1120 char1,0,11,v$
1130 char1,0,14,"Az elmozdulás értéke (m) : "
1140 a=all:p=27:q=14:gosub1620:all=a
1150 char1,0,17,"A munka"
1160 char1,5,19,"- kezdoerteke (J) : "
1170 a=kezd:p=25:q=19:gosub1620:kezd=a
1180 char1,5,21,"- lepeskoz (J) : "
1190 a=lep:p=22:q=21:gosub1620:lep=a
1200 col=9:gosub1860
1210 char1,18,1,"  ────▶"
1220 forj=0to9
1230 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1240 next
1250 gotol750
1260 :
1270 char1,25,5,v3$
1280 char1,25,7,v1$
1290 char1,25,9,v2$
1300 char1,0,11,v$
1310 char1,0,14,"A munka értéke (J) : "
1320 a=all:p=21:q=14:gosub1620:all=a
1330 char1,0,17,"Az éro"
1340 char1,5,19,"- kezdoerteke (N) : "
1350 a=kezd:p=25:q=19:gosub1620:kezd=a
1360 char1,5,21,"- lepeskoz (N) : "
1370 a=lep:p=22:q=21:gosub1620:lep=a
1380 col=9:gosub1860
1390 char1,28,1,"  ───────────▶"
1400 forj=0to9
1410 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1420 next
1430 gotol750
1440 :
1450 char1,25,5,v3$
1460 char1,25,7,v2$
1470 char1,25,9,v1$
1480 char1,0,11,v$
1490 char1,0,14,"Az éro értéke (N) : "
1500 a=all:p=20:q=14:gosub1620:all=a

```

```

1510 char1,0,17,"A munka"
1520 char1,5,19,"- kezdoerteke (J) : "
1530 a=kezd:p=25:q=19:gosub1620:kezd=a
1540 char1,5,21,"- lepeskoz (J) : "
1550 a=lep:p=22:q=21:gosub1620:lep=a
1560 col=9:gosub1860
1570 char1,28,1," ────────────"
1580 forj=0to9
1590 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1600 next
1610 goto1750
1620 :
1630 b$="":char1,p,q," "
1640 print" ←";getkeya$
1650 ifa$=chr$(13)then1690
1660 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$a$
1670 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1680 iflen(b$)<6then1640
1690 ifb$=""then1720
1700 ifval(b$)<.01thenchar1,p,q,sp$:goto1620
1710 a=val(b$)
1720 char1,p,q,str$(a)+sp$
1730 ifa<1thenchar1,p,q,"0"
1740 return
1750 :
1760 char1,0,0,f1$:char1,0,3,f2$
1770 forj=0to9
1780 char1,0,6+2*j," "
1790 printusing"#####.##";x(j);
1800 printusing"#####.##";y(j);
1810 printusing"#####.##";z(j);
1820 next
1830 getkeya$:ifa$=" "then480:elseifa$=""then320:elseifa$=chr$(14
0)thenrun
1840 goto1830
1850 stop
1860 :
1870 scnclr
1880 color0,col,6:color1,col,1:color4,col,6
1890 return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 95 s (31 mp.)

Speciális karakter: nincs

A program a teljesítmény fogalmának elmélyítéséhez használható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10féle eszközre vonatkozó adatok szerepelnek.

Az első oszlopban az eszköz neve, a másodikban a munka, a harmadikban a munkavégzés időtartama (kezdeti értéke 10 s) az utolsóban pedig a teljesítmény értéke található.

Az első adat előtt álló > jel a kurzormozgató billentyűvel (nyilakkal) bármelyik adat elé elvihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban jelenik meg a képernyőn (sötét alapon világos betűk).

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Ezzel az eljárással egy kiválasztott eszköz esetén a munka és a munkavégzés időtartama közti egyenes arányosság egyszerűen tanulmányozható.

Lehetőség van arra, hogy valamennyi munka, illetve időadatot egyszerre változtassuk. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja. Ezzel lehetőség nyílik azonos munkát végző (illetve azonos ideig dolgozó) eszközök munkavégzési idejének (illetve munkájának) az összehasonlítására.

A programban szereplő anyagok a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni az eszköz nevét, majd az idézőjelen kívülre, a névtől egy vesszővel elválasztva a

teljesítményt. Ügyeljünk arra, hogy a tizedesvessző helyett tizedespontot kell írni. Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A gép funkcióbillentyűi a program leállítása után is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ez a programot is törli a memóriából.

```
10 rem teljesitmeny (tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 rem fori=2to7:keyi,"":next
40 fori=1to7:keyi,"":next:key8,chr$(140)
50 scnclr:color0,9,2:color1,2,7:color4,9,2
60 printchr$(14);chr$(8);
70 print"  ";
80 print"  "v@b
90 print"  "____v/
100 print"  v(____)  TELJESITMENY (tabl.1.)"
110 print"  ||  ||"
120 print"  ||  ||"
130 print"  A programmal 10 kulonbozo test teljesit-"
140 print"menye tanulmanyozhato."
150 print"  A > jel a billentyuzeten talalhato"
160 print"nyilakkal a megfelelo helyre viheto."
170 print"  A kivant ertek beirasa utan a RETURN"
180 print"gomb megnyomasara a program atirja a"
190 print"tablazat adatait."
200 print"  A beirt ertek legfeljebb 4 karakterbol"
210 print"allhat."
220 print"  Az ESC billentyu megnyomasa utan az"
230 print"egesz oszlop erteke atirhato."
240 char1,25,24,"TOVABB =  "
250 fori=0to9:readnev$(i),z(i):next
260 getkeya$:ifa$=chr$(140)thenrun
270 ifa$<>"t"anda$<>"T"then260
280 color0,10,2:color1,2,7:color4,10,2
290 fori=0to9:y(i)=10:x(i)=y(i)*z(i):next
300 p=1:q=0
310 scnclr
320 print"          W          t          P"
330 print"          (J)          (s)          (W)"
340 fori=0to9:gosub520:next
350 b$=""
360 char1,12*p-2,4+2*q,">"
370 getkeya$:ifa$=chr$(140)thenrun
380 ifa$=" "andp=2andb$=""thenp=1:print" ":goto350
390 ifa$=" "andp=1andb$=""thenp=2:print" ":goto350
400 ifa$=" "andq>0andb$=""thenq=q-1:print" ":goto350
410 ifa$=" "andq<9andb$=""thenq=q+1:print" ":goto350
420 ifa$=" "andb$=""thenprint" ":p=1:q=0:goto350
430 ifa$=" "andb$=""thenp=2:fori=0to9:y(i)=10:gosub520:next:p=1:go
oto350
440 ifa$=chr$(27)andb$=""thengosub610:goto350
```

```

450 if(a$<"0"ora$>"9")anda$<>". "then370
460 i=q
470 print"
480 char1,12*p-2,4+2*i,"> "
490 gosub700
500 gosub770
510 goto350
520 :
530 ifp=1theny(i)=x(i)/z(i)
540 ifp=2thenx(i)=y(i)*z(i)
550 char1,0,4+2*i,nev$(i)
560 char1,10,4+2*i,"
570 printusing"##### ";x(i);
580 printusing" ##### ";y(i);
590 printusing" #####.#";z(i);
600 return
610 :
620 print"███ ":char1,12*p-2,24,"> "
630 gosub700
640 fori=0to9:gosub770:next
650 char1,12*p-2,24," "
660 q=0
670 return
680 :
690 getkeya$
700 ifa$=chr$(140)thenrun
710 ifa$=chr$(13)andb$>" "then760
720 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):print"███ ███";:g
oto690
730 if(a$>"9"ora$<"0")anda$<>". "then690
740 b$=b$+a$
750 print"███"+a$;" ";:iflen(b$)<4then690
760 return
770 :
780 b=val(b$)
790 ifp=1thenx(i)=b:elsey(i)=b
800 gosub520
810 return
820 :
830 data"Zsebradio",1
840 data"C= 16 gep",9
850 data"Junoszt tv",45
860 data"Ember",100
870 data"Csillar",300
880 data"Lo",700
890 data"Vasalo",1000
900 data"Mosogep",2200
910 data"Polски 126",11700
920 data"Villamos",60000

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 110 s (32 mp.)

Speciális karakter: @ → " £ → h [ → Δ

A program a hatások fogalmának elmélyítéséhez használható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10féle eszközre vonatkozó adatok szerepelnek.

Az első oszlopban az eszköz neve, a másodikban a hasznos munka, a harmadikban a befektetett munka (kezdeti értéke 10 cm), az utolsó sorban pedig a hatásfok értéke található.

Az első adat előtt álló > jel a kurzormozgató billentyűvel (nyilakkal) bármelyik adat elé elvihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN billentyű lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban jelenik meg a képernyőn (sötét alapon világos betűk).

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Ezzel az eljárással egy kiválasztott eszköz esetén a hasznos és a befektetett munka közötti egyenes arányosság egyszerűen tanulmányozható.

Lehetőség van valamennyi eszköz hasznos, illetve összes munkájának egyszerre történő megváltoztatására is. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja. Ezzel lehetőség nyílik azonos hasznos (illetve azonos összmunkát) végzett eszközök összes (illetve hasznos) munkájának összehasonlítására.

A programban szereplő anyagok a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni az eszköz nevét, majd az



idézőjelen kívülre, a névtől egy vesszővel elválasztva a hatásfokát. Ügyeljünk arra, hogy a tizedesvessző helyett tizedes-pontot kell írni. Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A karakterkészlet átírása miatt a képernyő a leállítás után elkövetett esetleges hiba miatt olvashatatlaná válhat. Ez a helyzet az F1 billentyű megnyomásával megszüntethető. Ezzel a módszerrel a program sem törlődik. A gép a program leállítása után is a módosított karakterkészletet használja, továbbá a funkció-billentyűk is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ezzel a program is törlődik.

```
10 rem hatasfok (tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scncrl
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next:rem @
110 fori=0to7:reada:poke15576+i,a:next:rem [
120 fori=0to7:reada:poke15584+i,a:next:rem £
130 poke65298,56:poke65299,56
140 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
150 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
160 data124,230,102,102,102,6,6,6
170 data8,20,20,36,34,66,126,0
180 data0,0,0,64,64,112,72,72
190 key1,"poke65298,56:poke65299,56"+chr$(13)
200 fori=2to7:keyi,"":next
210 key8,chr$(140)
220 color0,9,5:color4,9,5
230 printchr$(14);chr$(8)
240 print"
250 print" ← [E] @ [E] ← HATASFOK (tabl.1.)"
260 print" [E£ [E"
270 print" [A programmal 10 kulonbozo eszkoz hatas-"
280 print" foka tanulmanyozhato."
290 print" [A > jel a billentyuzeten talalhato"
300 print" nyilakkal a megfelelo helyre viheto."
310 print" [A kivant ertekek beirasa utan a RETURN"
320 print" gomb megnyomasara a program atirja a"
330 print" tablazat adatait."
340 print" [A beirt ertekek legfeljebb 4 karakterbol"
350 print" allhat."
360 print" [Az ESC billentyu megnyomasa utan az"
370 print" oszlop erteke atirhato."
380 char1,25,24,"TOVABB = [T]"
390 fori=0to9:readnev$(i),z(i):next
400 getkeya$:ifa$=chr$(140)thenrun
```

```

410 ifa$<>"t"anda$<>"T"then400
420 color0,10,2:color1,2,7:color4,10,2
430 fori=0to9:y(i)=10:x(i)=y(i)*z(i):next
440 p=1:q=0
450 scnclr
460 print"          [E£          [E          @"
470 print"          (J)          (J)          (%)"
480 fori=0to9:gosub660:next
490 b$=""
500 char1,10*p,4+2*q,">"
510 getkeya$:ifa$=chr$(140)thenrun
520 ifa$="!"andp=2andb$=""thenp=1:print".":goto490
530 ifa$="|"andp=1andb$=""thenp=2:print" ":goto490
540 ifa$="}"andq>0andb$=""thenq=q-1:print" ":goto490
550 ifa$="{"andq<9andb$=""thenq=q+1:print" ":goto490
560 ifa$="}"andb$=""thenprint" ":p=1:q=0:goto490
570 ifa$="{"andb$=""thenp=2:fori=0to9:y(i)=10:gosub660:next:p=1:g
oto490
580 ifa$=chr$(27)andb$=""thengosub750:goto490
590 if(a$<"0"ora$>"9")anda$<>". "then510
600 i=q
610 print"          "
620 char1,10*p,4+2*i,"> "
630 gosub840
640 gosub910
650 goto490
660 :
670 ifp=1theny(i)=x(i)/z(i)
680 ifp=2thenx(i)=y(i)*z(i)
690 char1,0,4+2*i,nev$(i)
700 char1,10,4+2*i,""
710 printusing"#####. # ";x(i);
720 printusing"#####. # ";y(i);
730 printusing"##### ";z(i)*100;
740 return
750 :
760 print"!!! ":char1,10*p,24,"> "
770 gosub840
780 fori=0to9:gosub910:next
790 char1,10*p,24," "
800 q=0
810 return
820 :
830 getkeya$
840 ifa$=chr$(140)thenrun
850 ifa$=chr$(13)andb$>" "then900
860 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):print"!!! !!!";g
oto830
870 if(a$>"9"ora$<"0")anda$<>". "then830
880 b$=b$+a$
890 print"!!!"+a$;" ";:iflen(b$)<4then830
900 return
910 :

```

```
920 b=val(b$)
930 ifp=1thenx(i)=b:elsey(i)=b
940 gosub660
950 return
960 :
970 data"Izzolampa",.01
980 data"Gozgep",.15
990 data"Gozturbina",.25
1000 data"Otto motor",.30
1010 data"Diesel m.",.40
1020 data"Csavar",.50
1030 data"Csigasor",.50
1040 data"Allocsiga",.80
1050 data"Vill.motor",.90
1060 data"Vill.futes",.95
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 200 s (40 mp.)  
 Speciális karakter: @ →  $\eta$  £ → h

A program az  $\eta = W_h / W_0$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit. /

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább egy menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, amelynek értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésköznek 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységeket nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól. Ezekután három lehetőség közül választhatunk. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN megnyomásával az előzőleg használt értéket megtartsuk.

A program futása a RUN/STOP billentyű megnyomásával szakítható meg. Mivel a gép a program leállítása után is a módosított karakterkészletet használja, egy esetleges hiba a képernyő olvashatatlaná válását eredményezi. Ez az állapot az f1 gomb megnyomásával megszüntethető. A funkcióbillentyűk átprogramozott állapota és a karakterkészlet a RESET gomb megnyomásával állítható vissza az eredeti állapotba, de ez a program törlését eredményezi.

```
10 rem hatasfok (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnc1r
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next:rem @
110 fori=0to7:reada:poke15584+i,a:next:rem £
120 poke65298,56:poke65299,56
130 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
140 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
150 data124,230,102,102,102,6,6,6
160 data0,0,0,32,32,56,36,36
170 key1,"poke65298,56:poke65299,56"+chr$(13)
180 fori=2to7:keyi,"":next:key8,chr$(140)
190 f1$="          W£          W          @"
200 f2$="          (J)          (J)          (%)"
210 v1$="HASZNOS MUNKA"
220 v2$="BEFEKTETETT MUNKA"
230 v3$="HATASFOK"
240 v$="-----"
250 sp$=" "
260 col=11:gosub2020
270 printchr$(14);chr$(8)
280 print"
290 print" ←  @  ← HATASFOK (tabl.2)"
300 print"
310 print" W£          W"
320 print" A program az @ = W£/ W összefugges elem-"
330 print" zesehez keszult."
340 print" A három mennyiség közül egyet allando"
350 print" értéknek választva egy másik mennyiség"
360 print" értéke szabadon változtatható."
370 print" A program ezek figyelembevételével"
380 print" kiszámítja a harmadik változó megfelelő"
390 print" értékeit."
400 print" A CLEAR/HOME megnyomására új értékek"
410 print" adhatók az egyes mennyiségeknek, a"
```

```

420 print"SHIFT es CLEAR/HOME egyutt es lenyomasa"
430 print"utan pedig az egyes mennyisegek szerepe"
440 print"valtoztathato meg."
450 char1,25,24,"TOVABB = T"
460 getkeya$:ifa$=chr$(140)thenrun
470 ifa$<"t"anda$<"T"then460
480 col=10:gosub2020
490 all=50:kezd=10:lep=1
500 print" " +sp$+ "          A VIZSGALT ESET KIVALASZTASA          "+sp$
510 char1,7,5,"FUGGO          FUGGETLEN          "
520 char1,18,6,"ALLANDO"
530 char1,6,7,"VALTOZO          VALTOZO          "
540 char1,0,8,v$
550 char1,0,9," 1.          W£          W          @          "+sp$
560 char1,0,11," 2.          W£          @          W          "+v$
570 char1,0,13," 3.          W          W£          @          "+sp$
580 char1,0,15," 4.          W          @          W£          "+v$
590 char1,0,17," 5.          @          W          W£          "+sp$
600 char1,0,19," 6.          @          W£          W          "+v$
610 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET          VALASZTA
NI."
620 getkeyt$:ift$=chr$(140)thenrun
630 ift$<"1"ort$>"6"then620
640 col=15:gosub2020
650 print" ";sp$;"          ERTEKADAS          ";sp$
660 char1,2,5,"FUGGO VALTOZO          : "
670 char1,2,7,"Allando mennyiseg          : "
680 char1,2,9,"Fuggetlen valtozo          : "
690 onval(t$)goto700,880,1060,1240,1420,1600
700 :
710 char1,22,5,v1$
720 char1,22,7,v2$
730 char1,22,9,v3$
740 char1,0,11,v$
750 char1,0,14,"A befektetett munka (J) : "
760 a=all:p=26:q=14:gosub1780:all=a
770 char1,0,17,"A hatasfok"
780 char1,5,19,"- kezdoerteke (%) : "
790 a=kezd:p=25:q=19:gosub1780:kezd=a
800 char1,5,21,"- lepeskoz (%) : "
810 a=lep:p=22:q=21:gosub1780:lep=a
820 col=9:gosub2020
830 char1,5,1," — "
840 forj=0to9
850 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)/100
860 next
870 goto1910
880 :
890 char1,22,5,v1$
900 char1,22,7,v3$
910 char1,22,9,v2$
920 char1,0,11,v$
930 char1,0,14,"A hatasfok erteke (%) : "

```

```

940 a=all:p=24:q=14:gosub1780:all=a
950 char1,0,17,"A befektetett munka"
960 char1,5,19,"- kezdoerteke (J) : "
970 a=kezd:p=25:q=19:gosub1780:kezd=a
980 char1,5,21,"- lepeskoz (J) : "
990 a=lep:p=22:q=21:gosub1780:lep=a
1000 col=9:gosub2020
1010 char1,5,1," ┌───┐"
1020 forj=0to9
1030 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)/100
1040 next
1050 goto1910
1060 :
1070 char1,22,5,v2$
1080 char1,22,7,v1$
1090 char1,22,9,v3$
1100 char1,0,11,v$
1110 char1,0,14,"A hasznos munka (J) : "
1120 a=all:p=22:q=14:gosub1780:all=a
1130 char1,0,17,"A hatasfok"
1140 char1,5,19,"- kezdoerteke (%) : "
1150 a=kezd:p=25:q=19:gosub1780:kezd=a
1160 char1,5,21,"- lepeskoz (%) : "
1170 a=lep:p=22:q=21:gosub1780:lep=a
1180 col=9:gosub2020
1190 char1,18,1," ┌───┐"
1200 forj=0to9
1210 x(j)=all:z(j)=kezd+j*lep:y(j)=all*100/z(j)
1220 next
1230 goto1910
1240 :
1250 char1,22,5,v2$
1260 char1,22,7,v3$
1270 char1,22,9,v1$
1280 char1,0,11,v$
1290 char1,0,14,"A hatasfok erteke (%) : "
1300 a=all:p=24:q=14:gosub1780:all=a
1310 char1,0,17,"A hasznos munka"
1320 char1,5,19,"- kezdoerteke (J) : "
1330 a=kezd:p=25:q=19:gosub1780:kezd=a
1340 char1,5,21,"- lepeskoz (J) : "
1350 a=lep:p=22:q=21:gosub1780:lep=a
1360 col=9:gosub2020
1370 char1,18,1," ┌───┐"
1380 forj=0to9
1390 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)*100/all
1400 next
1410 goto1910
1420 :
1430 char1,22,5,v3$
1440 char1,22,7,v1$
1450 char1,22,9,v2$
1460 char1,0,11,v$

```

```

1470 char1,0,14,"A hasznos munka (J) : "
1480 a=all:p=22:q=14:gosub1780:all=a
1490 char1,0,17,"A befektetett munka"
1500 char1,5,19,"- kezderteke (J) : "
1510 a=kezd:p=25:q=19:gosub1780:kezd=a
1520 char1,5,21,"- lepeskoz (J) : "
1530 a=lep:p=22:q=21:gosub1780:lep=a
1540 col=9:gosub2020
1550 char1,31,1," ┌───┐"
1560 forj=0to9
1570 x(j)=all:y(j)=kezd+j*lep:z(j)=all*100/y(j)
1580 next
1590 goto1910
1600 :
1610 char1,22,5,v3$
1620 char1,22,7,v2$
1630 char1,22,9,v1$
1640 char1,0,11,v$
1650 char1,0,14,"A befektetett munka (J) : "
1660 a=all:p=26:q=14:gosub1780:all=a
1670 char1,0,17,"A hasznos munka"
1680 char1,5,19,"- kezderteke (J) : "
1690 a=kezd:p=25:q=19:gosub1780:kezd=a
1700 char1,5,21,"- lepeskoz (J) : "
1710 a=lep:p=22:q=21:gosub1780:lep=a
1720 col=9:gosub2020
1730 char1,31,1," ┌───┐"
1740 forj=0to9
1750 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)*100/all
1760 next
1770 goto1910
1780 :
1790 b$="":char1,p,q," "
1800 print" ←┌┐┐┐";getkeya$
1810 ifa$=chr$(13)then1850
1820 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$+a$
1830 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1840 iflen(b$)<6then1800
1850 ifb$=""then1880
1860 ifval(b$)<.01thenchar1,p,q,sp$:goto1780
1870 a=val(b$)
1880 char1,p,q,str$(a)+sp$
1890 ifa<1thenchar1,p,q,"0"
1900 return
1910 :
1920 char1,0,0,f1$:char1,0,3,f2$
1930 forj=0to9
1940 char1,0,6+2*j," "
1950 printusing"#####.#";x(j);
1960 printusing"#####.#";y(j);
1970 printusing"#####.#";z(j);

```



```
1980 next
1990 getkeya$:ifa$="☐"then640:elseifa$=""then480:elseifa$=chr$(14
0)thenrun
2000 goto1990
2010 stop
2020 :
2030 scnclr
2040 color0,col,6:color1,col,1:color4,col,6
2050 return
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 110 s (32 mp.)

Speciális karakter: @ → μ £ → > [ → n

A program a súrlódási együttható fogalmának elmélyítéséhez használható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10 pár anyagra vonatkozó adatok szerepelnek. Az első oszlopban a két anyag neve, a másodikban a súrlódási erő, harmadikban a nyomóerő (kezdeti értéke 10 N), az utolsóban pedig a súrlódási tényező található.

Az első adat előtt álló > jel a kurzormozgató billentyűkkel (nyilakkal) bármelyik adat elé elvihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban jelenik meg a képernyőn (sötét alapon világos betűk).

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Ezzel az eljárással két megadott anyag esetén a súrlódási erő és a nyomóerő közti egyenes arányosság egyszerűen tanulmányozható.

Lehetőség van arra, hogy valamennyi súrlódási erőt (illetve nyomóerőt) egyszerre változtassuk meg. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja. Ezzel lehetőség nyílik azonos súrlódási erők (illetve azonos nyomóerők) esetén a nyomóerők (illetve a súrlódási erők) összehasonlítására.

A programban szereplő anyagok a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni a két anyag nevét, majd az idézőjelen kívülre, a névtől egy vesszővel elválasztva a súrlódási tényezőjüket. Ügyeljünk arra, hogy a tizedesvessző helyett

tizedespontot kell írni. Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A gép funkcióbillentyűi a program leállítása után is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ez a programot is törli a memóriából.

```
10 rem surlodas (tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 fori=0to7:reada:poke15584+i,a:next
120 fori=0to7:reada:poke15576+i,a:next
130 poke65298,56:poke65299,56
140 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
150 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
160 data0,51,51,51,115,222,192,128
170 data0,0,0,0,240,192,48,240
180 data0,0,0,0,224,144,144,144
190 key1,"poke65298,56:poke65299,56"+chr$(13)
200 fori=2to7:keyi,"":next
210 key8,chr$(140)
220 color0,2,6:color1,1,0:color4,2,6
230 printchr$(14);chr$(8)
240 print"
250 print" ← @ SURLODAS (tabl.1.)"
260 print"
270 print" A programmal 10 par kulonbozo anyag"
280 print"surlodasa tanulmanyozhato."
290 print" A > jel a billentyuzeten talalhato"
300 print"nyilakkal a megfelelo helyre viheto."
310 print" A kivant ertek beirasa utan a RETURN"
320 print"gomb megnyomasara a program atirja a"
330 print"tablazat adatait."
340 print" A beirt ertek legfeljebb 4 karakterbol"
350 print"allhat."
360 print" Az ESC billentyu megnyomasa utan az"
370 print"oszlop erteke atirhato."
380 char1,25,24,"TOVABB = T"
390 fori=0to9:readnev$(i),z(i):next
400 getkeya$:ifa$=chr$(140)thenrun
410 ifa$<>"t"anda$<>"T"then400
420 color0,10,2:color1,2,7:color4,10,2
430 fori=0to9:y(i)=10:x(i)=y(i)*z(i):next
```

```

440 p=1:q=0
450 scnclr
460 print"          F£          FL          @"
470 print"          (N)          (N)"
480 fori=0to9:gosub660:next
490 b$=""
500 char1,6+8.5*p,4+2*q,">"
510 getkeya$:ifa$=chr$(140)thenrun
520 ifa$="|"andp=2andb$=""thenp=1:print" ":goto490
530 ifa$="|"andp=1andb$=""thenp=2:print" ":goto490
540 ifa$="|"andq>0andb$=""thenq=q-1:print" ":goto490
550 ifa$="|"andq<9andb$=""thenq=q+1:print" ":goto490
560 ifa$="|"andb$=""thenprint" ":p=1:q=0:goto490
570 ifa$="|"andb$=""thenp=2:fori=0to9:y(i)=10:gosub660:next:p=1:g
oto490
580 ifa$=chr$(27)andb$=""thengosub750:goto490
590 if(a$<"0"ora$>"9")anda$<>". "then510
600 i=q
610 print"          "
620 char1,6+8.5*p,4+2*i,"> "
630 gosub840
640 gosub910
650 goto490
660 :
670 ifp=1theny(i)=x(i)/z(i)
680 ifp=2thenx(i)=y(i)*z(i)
690 char1,0,4+2*i,nev$(i)
700 char1,16,4+2*i," "
710 printusing"####.#";x(i);
720 printusing"#####";y(i);
730 printusing"###.### ";z(i);
740 return
750 :
760 print"|| " :char1,8+8*p,24,"> "
770 gosub840
780 fori=0to9:gosub910:next
790 char1,8+8*p,24," "
800 q=0
810 return
820 :
830 getkeya$
840 ifa$=chr$(140)thenrun
850 ifa$=chr$(13)andb$>" "then900
860 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):print"|| ||";:g
oto830
870 if(a$>"9"ora$<"0")anda$<>". "then830
880 b$=b$+a$
890 print"|" +a$;" " ;:iflen(b$)<4then830
900 return

```

```
910 :
920 b=val(b$)
930 ifp=1thenx(i)=b:elsey(i)=b
940 gosub660
950 return
960 :
970 data"Gumi-aszfalt",0.7
980 data"Gumi-aszf.+viz",0.4
990 data"Fa-fa",0.4
1000 data"Bor-fem",0.3
1010 data"Acel-vas",0.18
1020 data"Acel-acel",0.15
1030 data"Vas-ho",0.035
1040 data"Fa-jeg",0.035
1050 data"Acel-jeg",0.014
1060 data"Acel-acel+olaj",0.01
```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 200 s (40 mp.)

Speciális karakter: nincs

A program az  $M = F * k$  összefüggés feladatokban történő alkalmazását gyakoroltatja. A program elindítása után egy rövid ismertető jelenik meg. Ennek elolvasása után a T lenyomásával léphetünk tovább. A megjelenő menü alapján választhatjuk ki, hogy melyik mennyiség kiszámítását akarjuk gyakorolni. A választáshoz csak a megfelelő számot kell lenyomni. Itt nyílik lehetőség arra is, hogy a 9 gomb lenyomásával az addig elért eredményeket kiírassuk a képernyőre. A HELP megnyomásával a program elején található bevezetőhöz lehet visszalépni.

A megfelelő mennyiség kiválasztása után egy feladat jelenik meg, melynek szövege illetve adatai véletlenszerűen változnak. A HELP megnyomására a gép megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az elért eredménybe.

A választ a billentyűzet segítségével lehet beírni, és a RETURN billentyű megnyomásával kell lezárni. A válaszba a mértékegységet is be kell írni. A mértékegységeket rövidítve (Nm, N, m) vagy teljes nevükkel (newtonméter, newton, méter) is beírhatjuk.

A gép a választ elemzi, és hiba esetén annak jellegére utaló üzenet kiírása után megismétli a kérdést. Ha a második válasz is helytelen, a megoldást is kiírja a képernyőre. Helyes válasz esetén csak a HELYES felirat jelenik meg. Ezután az utolsó sorban megjelenő kiírásnak megfelelően az U billentyűvel új feladat kérhető, vagy az M gombbal a menühez térhetünk vissza, és más típusú feladatot választhatunk. A HELP billentyűvel ez esetben is a bevezetőhöz lehet visszamenni.

Az elért eredmények kiírását a menü alapján a 9 gomb lenyomásával kérhetjük. A képernyőn típusonként és összesítve is megjelenik a kapott feladatok száma, az elért pontszám és az eredmény (százalékban kifejezve). Egy feladatra 2 illetve 1 pont kapható aszerint, hogy az első vagy csak a második válasz volt a helyes. A T billentyűvel a menühez, a HELP segítségével pedig a bevezetőhöz lehet visszajutni.

A program a RUN/STOP billentyűvel állítható le. A leállítás után a funkcióbillentyűk átprogramozott állapotban maradnak. Ez a helyzet a RESET gomb megnyomásával szüntethető meg, de ez az eljárás a programot is törli.

```
10 rem forgatonyomatek (gyak.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 fori=1to9
40 readk$(i),t(i),p1(i),p2(i),d1(i),d2(i)
50 next
60 graphic0,1
70 fori=1to7:keyi,"":next:key8,chr$(140)
80 col=10:gosub940
90 printchr$(14);chr$(8)
100 print"  ↑"
110 print"  F | k          FORGATONYOMATEK (gyak.1.)"
120 print"      └──────────┘"
130 print"■■■■A program az  $M = F * k$  osszefugges fel-"
140 print"adatokban torteno alkalmazasat gyakorol-tatja."
150 print"■A valaszokba a mertekegysegeket is be  kell irni."
160 print"■A valaszt a RETURN gomb megnyomasaval  kell lezarni."
170 print"■A program jegyzi az elert eredmenyt."
180 char1,25,24,"TOVABB = ■T■"
190 getkeya$:ifa$="t"ora$="T"then200:elseifa$=chr$(140)then60:else
e190
200 scnc1r
210 col=7:gosub940
220 char1,3,3,"1. Forgatonyomatek |"
230 char1,21,4," |"
240 char1,3,5,"2. Erokar          > kiszamitasa"
250 char1,21,6," |"
260 char1,3,7,"3. Ero              |"
270 char1,3,9,"4. Vegyes feladatok"
280 char1,3,13,"5. Forgatonyomatek |"
290 char1,21,14," | kiszamitasa"
300 char1,3,15,"6. Erokar          >"
310 char1,21,16," | szoveg alapjan"
320 char1,3,17,"7. Ero              |"
330 char1,3,19,"8. Vegyes szoveges feladatok"
340 char1,3,23,"9. EREDMENYEK"
350 getkeyt$:ift$=chr$(140)then60
360 t=val(t$)
370 ift<1ort>9then350
380 ift=9then1610
390 ift=1theni=1+3*rnd(0)
400 ift=2theni=4+3*rnd(0)
410 ift=3theni=7+3*rnd(0)
420 ift=4thent=int(1+3*rnd(0)):goto390
430 ift=5theni=1+3*rnd(0):t=4
440 ift=6theni=4+3*rnd(0):t=5
450 ift=7theni=7+3*rnd(0):t=6
460 ift=8thent=int(5+3*rnd(0)):goto430
470 :
480 k$=k$(i):p1=p1(i):p2=p2(i):d1=d1(i):d2=d2(i)
```

```

490 scnc1r:col=11:gosub930
500 x=int(34+66*rnd(0))*10^(d1-2)
510 y=int(7*rnd(0)+3)*10^(d2-1)
520 z=x*y
530 x$=str$(x):ifx<1thenx$=" 0"+right$(x$,len(x$)-1)
540 y$=str$(y):ify<1theny$=" 0"+right$(y$,len(y$)-1)
550 z$=str$(z):ifz<1thenz$=" 0"+right$(z$,len(z$)-1)
560 r=0
570 ontgosub980,1030,1080,1130,1190,1250
580 char1,7,7,"←"
590 a$="":gosub1320
600 ifa$=chr$(140)thenprint" ":r(t)=r(t)-r:goto800
610 h=0
620 ifabs(val(b$)-e)>0.0001thenh=1
630 n=1000000
640 ifabs(val(b$)-e*n)<0.0001andn<>1thenh=2
650 n=n/10:ifn>.0000001then640
660 lb=len(b$):l1=len(m1$):l2=len(m2$)
670 ifright$(b$,l1)<>m1$then690
680 iflb>l1+1thenifval(left$(b$,lb-l1-1))=val(b$)then710:else720
690 ifright$(b$,l2)<>m2$then710
700 iflb>l2+1thenifval(left$(b$,lb-l2-1))=val(b$)then710:else720
710 h=h+4
720 char1,0,10,""
730 if(hand1)=1thenprint" Szamitasi hiba"
740 if(hand2)=2thenprint" Nagysagrendi hiba"
750 if(hand4)=4thenprint" Mertekegyseg hiba"
760 ifh=0then890
770 r=r+1:r(t)=r(t)+1
780 tt=2000:gosub1410
790 ifr=1thenscnc1r:goto570
800 char1,0,10,chr$(27)+"q"
810 char1,0,12,chr$(27)+"q"
820 ontgosub1430,1490,1550,1430,1490,1550
830 char1,7,24,"UJ FELADAT =  U  MENU =  M"
840 getkeya$:ifa$=chr$(140)then60
850 ifa$="u"ora$="U"then360
860 ifa$="m"ora$="M"then200
870 goto840
880 :
890 j(t)=j(t)+2-r
900 char1,7,14,"Helyes."
910 tt=1000:gosub1410
920 goto830
930 :
940 :
950 color0,col,1:color1,col,7:color4,col,1
960 return
970 :
980 char1,1,1,"F="+x$+" N"
990 char1,1,3,"k="+y$+" m"

```



```

1000 char1,1,5, "—————"
1010 char1,1,7, "M = ? " :gotol170
1020 return
1030 char1,1,1, "M =" +z$+ " Nm"
1040 char1,1,3, "F =" +x$+ " N"
1050 char1,1,5, "—————"
1060 char1,1,7, "k = ? " :gotol1230
1070 return
1080 char1,1,1, "M =" +z$+ " Nm"
1090 char1,1,3, "k =" +y$+ " m"
1100 char1,1,5, "—————"
1110 char1,1,7, "F = ? " :gotol1290
1120 return
1130 char1,0,1,k$
1140 char1,0,5, "Mekkora a forgatonyomatek?"
1150 char1,p1,1,x$
1160 char1,p2,1,y$
1170 e=z:m1$="Nm" :m2$="newtonmeter"
1180 return
1190 char1,0,1,k$
1200 char1,0,5, "Mekkora az erokar?"
1210 char1,p1,1,z$
1220 char1,p2,1,x$
1230 e=y:m1$="m" :m2$="meter"
1240 return
1250 char1,0,1,k$
1260 char1,0,5, "Mekkora az ero?"
1270 char1,p1,1,z$
1280 char1,p2,1,y$
1290 e=x:m1$="N" :m2$="newton"
1300 return
1310 :
1320 b$=""
1330 getkeya$:ifa$=chr$(140)thenreturn
1340 ifa$=chr$(13)andb$<>" "thenprint " ":return
1350 ifa$=chr$(20)andb$>" "thenprinta$;"← ████";b$=left$(b$,len(b$)
-1):gotol1330
1360 if(a$<="9"anda$>="0")or(a$<="z"anda$>="@")or(a$<="Z"anda$>="
✓")then1390
1370 ifa$="."ora$="*"ora$="/"then1390
1380 gotol1330
1390 printa$;"← ████";b$=b$+a$:iflen(b$)>15thenprint " ":return
1400 gotol1330
1410 :
1420 fortt=ttto0step-1:next:return
1430 :
1440 char1,7,12, "M = F * k"
1450 char1,7,14, "M =" +x$+ " N *" +y$+ " m"
1460 char1,7,16, "M =" +z$+ " N*m"
1470 char1,7,18, "M =" +z$+ " Nm"
1480 return
1490 :

```

```

1500 char1,7,12,"k = M / F"
1510 char1,7,14,"k = "+z$+" Nm / "+x$+" N"
1520 char1,7,16,"k = "+y$+" Nm/N"
1530 char1,7,18,"k = "+y$+" m"
1540 return
1550 :
1560 char1,7,12,"F = M / k"
1570 char1,7,14,"F = "+z$+" Nm / "+y$+" m"
1580 char1,7,16,"F = "+x$+" Nm/m"
1590 char1,7,18,"F = "+x$+" N"
1600 return
1610 :
1620 scnc1r
1630 col=2:gosub940
1640 char1,17,0,"feladat pont eredmeny"
1650 char1,0,3,"Forgatonyomatek"
1660 char1,0,5,"Erokar"
1670 char1,0,7,"Ero"
1680 char1,0,11,"Forgatonyom. (szov.)"
1690 char1,0,13,"Erokar (szov.)"
1700 char1,0,15,"Ero (szov.)"
1710 char1,0,19,"OSSZESEN"
1720 fori=1to3
1730 j=j(i):n=j+r(i)
1740 char1,19,1+2*i,"":printusing"###";n/2
1750 char1,26,1+2*i,"":printusing"###";j
1760 ifn=0then1790
1770 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1780 char1,34,1+2*i,"":printusing"### %";j*100/n
1790 color1,2:next
1800 fori=4to6
1810 j=j(i):n=j+r(i)
1820 char1,19,3+2*i,"":printusing"###";n/2
1830 char1,26,3+2*i,"":printusing"###";j
1840 ifn=0then1870
1850 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1860 char1,34,3+2*i,"":printusing"### %";j*100/n
1870 color1,2:next
1880 color1,8,6
1890 j=0:r=0:fori=1to6:j=j+j(i):r=r+r(i):next:n=r+j
1900 char1,19,19,"":printusing"###";n/2
1910 char1,26,19,"":printusing"###";j
1920 ifn=0then1940
1930 char1,34,19,"":printusing"### %";j*100/n
1940 color1,2
1950 gotol80

```

1960 data"Egy kilincset .. N erovel nyomunk .... m tavolsagban a  
 forgasponttol.  
 1970 data1,13,33,2,-1  
 1980 data"Egy hajtokarra .. N ero hat ... m tavol a forgastengel  
 ytol.  
 1990 data1,14,27,2,0  
 2000 data"Egy . m-es rud egyik veget ... N erovel emeljuk, masik  
 vege a foldon van."  
 2010 data1,26,3,3,1  
 2020 data"Egy kormanykerekre N erovel, Nm forgatonyomate  
 kkal hatunk."  
 2030 data2,31,18,2,0  
 2040 data"Egy fogaskerek lanca N erovel Nm forgatonyomate  
 kot hoz letre."  
 2050 data2,32,20,2,-1  
 2060 data"A merlegen levo N suly Nm forga-tonyomatekot f  
 ejt ki a merlegre."  
 2070 data2,25,15,2,0  
 2080 data"Egy m hosszú feszítovassal Nm forgatonyomate  
 kot fejtünk ki."  
 2090 data3,30,3,2,0  
 2100 data"A forgasponttol m-re Nm forga-tonyomatekkal  
 hatunk egy testre."  
 2110 data3,24,15,3,0  
 2120 data"Egy daru m-es karjara Nm forga-tonyomatekkal  
 hat a teher."  
 2130 data3,23,8,4,1

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 185 s (39 mp.)

Speciális karakter: nincs

A program az  $M = F * k$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább egy menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, amelynek értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésköznek 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységet nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezekután három lehetőség közül választhatunk. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN megnyomásával az előzőleg használt értéket megtartsuk. A program futása a RUN/STOP billentyű megnyomásával szakítható meg. A funkcióbillentyűk azonban a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez egyúttal a program törlését is jelenti.

```

10 rem forgatonyomatek (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 fori=1to7:key1,"":next:key8,chr$(140)
40 f1$="FORGATONYOMATEK   ERO           EROKAR"
50 f2$="      (Nm)           (N)           (m)"
60 v1$="FORGATONYOMATEK"
70 v2$="ERO"
80 v3$="EROKAR"
90 v$="-----"
100 sp$=" "
110 col=8:gosub1870
120 printchr$(14);chr$(8)
130 print"  ↑"
140 print" F|           FORGATONYOMATEK (tabl.2)"
150 print"  |     k"
160 print"  |_____o"
170 print"☐☐☐A program az M = F * k osszefugges"
180 print"elemzesehez keszult."
190 print"☐A harom mennyiseg kozul egyet allando"
200 print"ertekunek valasztva egy masik mennyiseg"
210 print"erteket szabadon valtoztathato."
220 print"☐A program ezek figyelembevetelevel"
230 print"kiszamitja a harmadik valtozo megfelelo"
240 print"ertekeit."
250 print"☐A CLEAR/HOME megnyomasara uj ertekek"
260 print"adhatok az egyes mennyisegeknek, a"
270 print"SHIFT es CLEAR/HOME egyuttles lenyomasa"
280 print"utan pedig az egyes mennyisegek szerepe"
290 print"valtoztathato meg."
300 char1,25,24,"TOVABB = ☐T☐"
310 getkeya$:ifa$=chr$(140)thenrun
320 ifa$<>"t"anda$<>"T"then310
330 col=10:gosub1870
340 all=10:kezd=1:lep=.5
350 print"☐"+sp$+"          A VIZSGALT ESET KIVALASZTASA          "+sp$
360 char1,7,5,"FUGGO           FUGGETLEN  "
370 char1,18,6,"ALLANDO"
380 char1,6,7,"VALTOZO           VALTOZO  "
390 char1,0,8,v$
400 char1,0,9," 1. Forg.nyom.   Ero           Erokar           "+sp$
410 char1,0,11," 2. Forg.nyom.   Erokar           Ero           "+v$
420 char1,0,13," 3. Ero           Forg.nyom.   Erokar           "+sp$
430 char1,0,15," 4. Ero           Erokar           Forg.nyom.   "+v$
440 char1,0,17," 5. Erokar           Forg.nyom.   Ero           "+sp$
450 char1,0,19," 6. Erokar           Ero           Forg.nyom.   "+v$

```

460 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET  
NI."

VALASZTA

470 getkeyt\$:ift\$=chr\$(140)thenrun

480 ift\$<"1"ort\$>"6"then470

490 col=15:gosub1870

500 print"☐";sp\$;" ERTEKADAS

";sp\$

510 char1,2,5,"FUGGO VALTOZO :"

520 char1,2,7,"Allando mennyiseg :"

530 char1,2,9,"Fuggetlen valtozo :"

540 onval(t\$)goto550,730,910,1090,1270,1450

550 :

560 char1,22,5,v1\$

570 char1,22,7,v2\$

580 char1,22,9,v3\$

590 char1,0,11,v\$

600 char1,0,14,"Az ero erteke (N) : "

610 a=all:p=20:q=14:gosub1630:all=a

620 char1,0,17,"Az erokar"

630 char1,5,19,"- kezdoerteke (m) : "

640 a=kezd:p=25:q=19:gosub1630:kezd=a

650 char1,5,21,"- lepeskoz (m) : "

660 a=lep:p=22:q=21:gosub1630:lep=a

670 col=9:gosub1870

680 char1,0,1,"☐—————☐"

690 forj=0to9

700 y(j)=all:z(j)=kezd+j\*lep:x(j)=all\*z(j)

710 next

720 goto1760

730 :

740 char1,22,5,v1\$

750 char1,22,7,v3\$

760 char1,22,9,v2\$

770 char1,0,11,v\$

780 char1,0,14,"Az erokar erteke (m) : "

790 a=all:p=23:q=14:gosub1630:all=a

800 char1,0,17,"Az ero"

810 char1,5,19,"- kezdoerteke (N) : "

820 a=kezd:p=25:q=19:gosub1630:kezd=a

830 char1,5,21,"- lepeskoz (N) : "

840 a=lep:p=22:q=21:gosub1630:lep=a

850 col=9:gosub1870

860 char1,0,1,"☐—————☐"

870 forj=0to9

880 z(j)=all:y(j)=kezd+j\*lep:x(j)=all\*y(j)

890 next

900 goto1760

910 :

920 char1,22,5,v2\$

930 char1,22,7,v1\$

940 char1,22,9,v3\$

950 char1,0,11,v\$

960 char1,0,14,"A forgatonyomatek (Nm) : "

970 a=all:p=25:q=14:gosub1630:all=a

```

980 char1,0,17,"Az erokar"
990 char1,5,19,"- kezdoerteke (m) : "
1000 a=kezd:p=25:q=19:gosub1630:kezd=a
1010 char1,5,21,"- lepeskoz (m) : "
1020 a=lep:p=22:q=21:gosub1630:lep=a
1030 col=9:gosub1870
1040 char1,18,1,"  ────  "
1050 forj=0to9
1060 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1070 next
1080 goto1760
1090 :
1100 char1,22,5,v2$
1110 char1,22,7,v3$
1120 char1,22,9,v1$
1130 char1,0,11,v$
1140 char1,0,14,"Az erokar erteke (m) : "
1150 a=all:p=23:q=14:gosub1630:all=a
1160 char1,0,17,"A forgatonyomatek"
1170 char1,5,19,"- kezdoerteke (Nm) : "
1180 a=kezd:p=26:q=19:gosub1630:kezd=a
1190 char1,5,21,"- lepeskoz (Nm) : "
1200 a=lep:p=23:q=21:gosub1630:lep=a
1210 col=9:gosub1870
1220 char1,18,1,"  ────  "
1230 forj=0to9
1240 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1250 next
1260 goto1760
1270 :
1280 char1,22,5,v3$
1290 char1,22,7,v1$
1300 char1,22,9,v2$
1310 char1,0,11,v$
1320 char1,0,14,"A forgatonyomatek (Nm) : "
1330 a=all:p=25:q=14:gosub1630:all=a
1340 char1,0,17,"Az ero"
1350 char1,5,19,"- kezdoerteke (N) : "
1360 a=kezd:p=25:q=19:gosub1630:kezd=a
1370 char1,5,21,"- lepeskoz (N) : "
1380 a=lep:p=22:q=21:gosub1630:lep=a
1390 col=9:gosub1870
1400 char1,28,1,"  ────────────  "
1410 forj=0to9
1420 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1430 next
1440 goto1760
1450 :
1460 char1,22,5,v3$
1470 char1,22,7,v2$
1480 char1,22,9,v1$
1490 char1,0,11,v$
1500 char1,0,14,"Az ero erteke (N) : "

```

```

1510 a=all:p=20:q=14:gosub1630:all=a
1520 char1,0,17,"A forgatonyomatek"
1530 char1,5,19,"- kezdoerteke (Nm) : "
1540 a=kezd:p=26:q=19:gosub1630:kezd=a
1550 char1,5,21,"- lepeskoz (Nm) : "
1560 a=lep:p=23:q=21:gosub1630:lep=a
1570 col=9:gosub1870
1580 char1,28,1,"  ────────────  "
1590 forj=0to9
1600 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1610 next
1620 goto1760
1630 :
1640 b$=" ":char1,p,q," "
1650 print" ←■■■■";:getkeya$
1660 ifa$=chr$(13)then1700
1670 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$+a$
1680 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1690 iflen(b$)<6then1650
1700 ifb$=""then1730
1710 ifval(b$)<0.01thenchar1,p,q,sp$:goto1630
1720 a=val(b$)
1730 char1,p,q,str$(a)+sp$
1740 ifa<1thenchar1,p,q,"0"
1750 return
1760 :
1770 char1,0,0,f1$:char1,0,3,f2$
1780 forj=0to9
1790 char1,0,6+2*j," "
1800 printusing"#####.##";x(j);
1810 printusing"#####.##";y(j);
1820 printusing"#####.##";z(j);
1830 next
1840 getkeya$:ifa$="▣"then490:elseifa$=""then330:elseifa$=chr$(14
0)thenrun
1850 goto1840
1860 stop
1870 :
1880 scnclr
1890 color0,col,6:color1,col,1:color4,col,6
1900 return

```



	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 200 s (40 mp.)

Speciális karakter: @ → z

A program a  $p = F / A$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább egy menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni, és melyik legyen a függő változó, amelynek értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésközt 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységeket nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezekután három lehetőség közül lehet választani. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN megnyomásával az előzőleg használt értéket megtartsuk.

A program futása a RUN/STOP billentyű megnyomásával szakítható meg. Mivel a gép a program leállítása után is a módosított karakterkészletet használja, egy esetleges hiba a képernyő olvashatatlaná válását eredményezi. Ez az állapot az f1 gomb megnyomásával megszüntethető. A funkcióbillentyűk átprogramozott állapota és a karakterkészlet a RESET gomb megnyomásával állítható vissza az eredeti állapotba, de ez a program törlését eredményezi.

```
10 rem nyomas (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 for i=0 to 21:read a:poke3048+i,a:next
80 for i=0 to 18:read a:poke4072+i,a:next
90 sys3048
100 for i=0 to 7:read a:poke15360+i,a:next:rem @
120 poke65298,56:poke65299,56
130 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
140 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
150 data56,36,12,16,60,0,0,0
170 key1,"poke65298,56:poke65299,56"+chr$(13)
180 for i=2 to 7:key i,"":next:key8,chr$(140)
190 f1$="          ERO          FELULET          NYOMAS  "
200 f2$="          (N)          (m0)          (Pa)  "
210 v1$="ERO"
220 v2$="FELULET"
230 v3$="NYOMAS"
240 v$="-----"
250 sp$=" "
260 col=11:gosub2010
270 printchr$(14);chr$(8);
280 print"  A  F"
290 print"  A  F"
300 print"  A  F----- NYOMAS (tabl.2.)"
303 print"  A  F"
310 print"  A  F"
315 print"  A program a p = F / A összefugges elem-"
320 print"zesehez keszult."
330 print"  A harom mennyiseg kozul egyet allando"
340 print"ertekunek valasztva egy masik mennyiseg"
350 print"erteke szabadon valtoztathato."
360 print"  A program ezek figyelembeveteleval"
370 print"kiszamitja a harmadik valtozo megfelelo"
380 print"ertekeit."
390 print"  A CLEAR/HOME megnyomasara uj ertekek"
400 print"adhatok az egyes mennyisegeknek, a"
```

```

410 print"SHIFT es CLEAR/HOME egyutt es lenyomasa"
420 print"utan pedig az egyes mennyisegek szerepe"
430 print"valtoztathato meg."
440 char1,25,24,"TOVABB =  T"
450 getkeya$:ifa$=chr$(140)thenrun
460 ifa$<"t"anda$<"T"then450
470 col=10:gosub2010
480 all=50:kezd=10:lep=1
490 print"  " +sp$+ "          A VIZSGALT ESET KIVALASZTASA          "+sp$
500 char1,7,5,"FUGGO          FUGGETLEN  "
510 char1,18,6,"ALLANDO"
520 char1,6,7,"VALTOZO          VALTOZO  "
530 char1,0,8,v$
540 char1,0,9," 1.   ERO          FELULET   NYOMAS          "+sp$
550 char1,0,11," 2.   ERO          NYOMAS    FELULET          "+v$
560 char1,0,13," 3.   FELULET   ERO        NYOMAS          "+sp$
570 char1,0,15," 4.   FELULET   NYOMAS    ERO          "+v$
580 char1,0,17," 5.   NYOMAS    ERO        FELULET          "+sp$
590 char1,0,19," 6.   NYOMAS    FELULET   ERO          "+v$
600 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET          VALASZTA
NI."
610 getkeyt$:ift$=chr$(140)thenrun
620 ift$<"1"ort$>"6"then610
630 col=15:gosub2010
640 print"  " ;sp$;"          ERTEKADAS          " ;sp$
650 char1,2,5,"FUGGO VALTOZO          :"
660 char1,2,7,"Allando mennyiseg          :"
670 char1,2,9,"Fuggetlen valtozo          :"
680 onval(t$)goto690,870,1050,1230,1410,1590
690 :
700 char1,22,5,v1$
710 char1,22,7,v2$
720 char1,22,9,v3$
730 char1,0,11,v$
740 char1,0,14,"A felulet erteke (m@)          :"
750 a=all:p=25:q=14:gosub1770:all=a
760 char1,0,17,"A nyomas"
770 char1,5,19,"- kezdoerteke (Pa)          :"
780 a=kezd:p=26:q=19:gosub1770:kezd=a
790 char1,5,21,"- lepeskoz (Pa)          :"
800 a=lep:p=23:q=21:gosub1770:lep=a
810 col=9:gosub2010
820 char1,6,1,"  —  "
830 forj=0to9
840 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
850 next
860 goto1900
870 :

```

```

880 char1,22,5,v1$
890 char1,22,7,v3$
900 char1,22,9,v2$
910 char1,0,11,v$
920 char1,0,14,"A nyomas erteke (Pa) : "
930 a=all:p=23:q=14:gosub1770:all=a
940 char1,0,17,"A felulet"
950 char1,5,19,"- kezdoerteke (m@) : "
960 a=kezd:p=27:q=19:gosub1770:kezd=a
970 char1,5,21,"- lepeskoz (m@) : "
980 a=lep:p=24:q=21:gosub1770:lep=a
990 col=9:gosub2010
1000 char1,6,1,"  ────"
1010 forj=0to9
1020 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
1030 next
1040 goto1900
1050 :
1060 char1,22,5,v2$
1070 char1,22,7,v1$
1080 char1,22,9,v3$
1090 char1,0,11,v$
1100 char1,0,14,"Az ero erteke (N) : "
1110 a=all:p=20:q=14:gosub1770:all=a
1120 char1,0,17,"A nyomas"
1130 char1,5,19,"- kezdoerteke (Pa) : "
1140 a=kezd:p=26:q=19:gosub1770:kezd=a
1150 char1,5,21,"- lepeskoz (Pa) : "
1160 a=lep:p=23:q=21:gosub1770:lep=a
1170 col=9:gosub2010
1180 char1,17,1,"  ────"
1190 forj=0to9
1200 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1210 next
1220 goto1900
1230 :
1240 char1,22,5,v2$
1250 char1,22,7,v3$
1260 char1,22,9,v1$
1270 char1,0,11,v$
1280 char1,0,14,"A nyomas erteke (Pa) : "
1290 a=all:p=23:q=14:gosub1770:all=a
1300 char1,0,17,"Az ero"
1310 char1,5,19,"- kezdoerteke (N) : "
1320 a=kezd:p=25:q=19:gosub1770:kezd=a
1330 char1,5,21,"- lepeskoz (N) : "
1340 a=lep:p=22:q=21:gosub1770:lep=a
1350 col=9:gosub2010

```

```

1360 char1,17,1," ────"
1370 forj=0to9
1380 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1390 next
1400 goto1900
1410 :
1420 char1,22,5,v3$
1430 char1,22,7,v1$
1440 char1,22,9,v2$
1450 char1,0,11,v$
1460 char1,0,14,"Az ero erteke (N) : "
1470 a=all:p=20:q=14:gosub1770:all=a
1480 char1,0,17,"A felulet"
1490 char1,5,19,"- kezdoerteke (m@) : "
1500 a=kezd:p=27:q=19:gosub1770:kezd=a
1510 char1,5,21,"- lepeskoz (m@) : "
1520 a=lep:p=24:q=21:gosub1770:lep=a
1530 col=9:gosub2010
1540 char1,30,1," ────"
1550 forj=0to9
1560 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1570 next
1580 goto1900
1590 :
1600 char1,22,5,v3$
1610 char1,22,7,v2$
1620 char1,22,9,v1$
1630 char1,0,11,v$
1640 char1,0,14,"A felulet erteke (m@) : "
1650 a=all:p=25:q=14:gosub1770:all=a
1660 char1,0,17,"Az ero"
1670 char1,5,19,"- kezdoerteke (N) : "
1680 a=kezd:p=25:q=19:gosub1770:kezd=a
1690 char1,5,21,"- lepeskoz (N) : "
1700 a=lep:p=22:q=21:gosub1770:lep=a
1710 col=9:gosub2010
1720 char1,30,1," ────"
1730 forj=0to9
1740 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1750 next
1760 goto1900
1770 :
1780 b$="":char1,p,q,""
1790 print" ←";getkeya$
1800 ifa$=chr$(13)then1840
1810 ifa$="."or(a$<="9"anda$>="0")thenprinta$;b$=b$a$
1820 ifa$=chr$(20)andb$>" "thenprinta$;b$=left$(b$,len(b$)-1)
1830 iflen(b$)<6then1790

```

```
1840 ifb$="" then1870
1850 ifval(b$)<0.01thenchar1,p,q,sp$:gotol770
1860 a=val(b$)
1870 char1,p,q,str$(a)+sp$
1880 ifa<1thenchar1,p,q,"0"
1890 return
1900 :
1910 char1,0,0;f1$:char1,0,3,f2$
1920 forj=0to9
1930 char1,0,6+2*j,""
1940 printusing"#####.#";x(j);
1950 printusing"#####.#";y(j);
1960 printusing"#####.#";z(j);
1970 next
1980 getkeya$:ifa$="☐" then630:elseifa$="" then470:elseifa$=chr$(14
0)thenrun
1990 gotol980
2000 stop
2010 :
2020 scnclr
2030 color0,col,6:color1,col,1:color4,col,6
2040 return
```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 80 s (29 mp.)

Speciális karakter: nincs

A programmal egy folyadékba helyezett szilárd test viselkedése tanulmányozható. A folyadék és a test sűrűsége is változtatható, így az úszás, a lebegés és a test lemerülése is megfigyelhető.

Az indítás után megjelenő rövid ismertető elolvasása után a T megnyomásával léphetünk tovább. A program ekkor felrajzolja a folyadékfelszínt és a folyadékban úszó szilárd testet, valamint kiírja a két anyag sűrűségét ( $0,5 \text{ g/cm}^3$  illetve  $1 \text{ g/cm}^3$ ). A képernyő alján pedig megjelenik az ÚSZIK felirat és az  $F_f = G$  összefüggés.

A ↑ és ↓ gombokkal ezután a test sűrűsége, a + és a - billentyűkkel a folyadék sűrűsége növelhető illetve csökkenthető. Mindkettő  $0,1 \text{ g/cm}^3$  és  $10 \text{ g/cm}^3$  közti értékeket vehet fel. A sűrűségek pillanatnyi értékét a program folyamatosan kiírja a képernyőre és ezzel egyidejűleg a megfelelő helyzetben megrajzolja a testet jelképező téglalapot. A képernyő alá kiírja, hogy a test úszik, lebeg vagy elmerült, illetve jelzi a felhajtóerő és a test súlya közti viszonyt.

A CLEAR/HOME megnyomásával a kezdeti sűrűségértékek állíthatók be. A HELP megnyomásával a bevezetőhöz mehetünk vissza. A program a RUN/STOP gombbal állítható le. Ha a villogó kurzor és a READY üzenet nem jelenne meg, mert a leállítás grafikus üzemmódban történt, a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN billentyű lenyomásával térhetünk vissza a normál üzemmódban.

A funkcióbillentyűk a program leállítása után is átprogramozott állapotban maradnak. Ez a helyzet a RESET megnyomásával megszüntethető, de ez egyúttal a programot is törli a memóriából.

```

10 rem uszas
20 rem zatonyi sandor bekescsaba 1986.
30 poke740,212:graphic0,1:color0,2,7:color4,7,6
40 fori=1to7:keyi,"":next:key8,chr$(8)
50 printchr$(14);chr$(8)
60 print"  A_mM"
70 print"  A_mM"
80 print"  U_____  U S Z A S"
90 print"  "
100 print"  A program segitsegevel egy szilard"
110 print" test folyadekban valo viselkedese      vizsgalható."
120 print"  A test surusege a felfele es lefele"
130 print" mutato nyilakkal, a folyadek surusege"
140 print" a + es - gombokkal változtathato."
150 print"  A suruseg erteke 0.1 g/cm3 es 10 g/cm3 kozott lehet."
160 print"  A CEAR/HOME megnyomasara az surusegek      a kiindulo er
tekeket veszik fel."
170 char1,25,23,"TOVABB = T"
180 getkeya$:ifa$<"t"anda$<"T"then180
190 graphic1,1:color1,1
200 fori=0to4:readn:poke8192+i,n:next
210 data 224,16,224,16,224
220 sshapek$,0,0,3,4:char1,0,0," "
230 draw1,0,100to320,100
240 gshapek$,194,7:gshapek$,226,23
250 char1,32,20,"F = G":char1,33,21,"f"
260 t=5:f=10
270 j=1:gosub370
280 getkeya$:ifa$=chr$(8)thenrun
290 ifa$=" "thenj=0:gosub370:goto260
300 ifa$="+"andf<100thenf=f+1:goto350
310 ifa$="-"andf>1thenf=f-1:goto350
320 ifa$=" "andt<100thent=t+1:goto350
330 ifa$=" "andt>1thent=t-1:goto350
340 goto280
350 j=0:gosub440:goto270
360 :
370 x=60*t/f:a$="USZIK ="
380 ifx>60thenx=99:a$="ELMERUL<":elseifx=60thenx=80:a$="LEBEG ="
390 char1,1,20,left$(a$,7):char1,35,20,right$(a$,1)
400 a$=str$(t/10)+" ":ift<10thena$=" 0"+right$(a$,len(a$)-1)
410 char1,10,1,"test:"+left$(a$,4)+" g/cm"
420 a$=str$(f/10)+" ":iff<10thena$=" 0"+right$(a$,len(a$)-1)
430 char1,10,3,"folyadek:"+left$(a$,4)+" g/cm"
440 boxj,115,40+x,205,100+x:draw1,115,100to205,100
450 return

```



	osztály	csoport	egyéni
új ismeret	●	●	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 80 s (29 mp.)

Speciális karakter: nincs

A program két test termikus kölcsönhatásának szimulálására alkalmas. A két test tömege és hőmérséklete - bizonyos korlátok közt - szabadon megadható, a két test fajhőjét viszont egyenlőnek tekinti a program. Az indítás után a többi programtól eltérően - a kis memóriaméret miatt - ismertető nem jelenik meg, hanem azonnal a tömegek majd a hőmérséklet értékét kéri a program. A tömegek maximális értéke 20 kg lehet.

A hőmérsékleteknek, hogy a tankönyv vízzel végzett kísérleteivel összhangban legyen a program,  $0^{\circ}\text{C}$  és  $100^{\circ}\text{C}$  között kell lenni. Ha egy beírt érték négy karakternél rövidebb, a bevittet a RETURN billentyűvel kell lezárni.

Az adatok beírása után a program sematikus felrajzolja a két, egymással érintkező testet egy-egy hőmérővel, és a hőmérséklet - idő grafikonhoz szükséges rendszert. A képernyőn látható feliratnak megfelelően az S gomb megnyomására (Start) folytatódik a program. A gép szimulálja a két test hőmérsékletének változását, és a pillanatnyi értékeket kiírja, illetve a hőmérőkön valamint a grafikonon is ábrázolja.

Ha kialakult a közös hőmérséklet, akkor ezt a grafikon végéhez is kiírja a program. Ellenkező esetben a T gomb megnyomásával tovább folytathatjuk a rajzoltatást, vagy előbb az A megnyomásával megnézhetjük a kiindulási adatokat, és csak ezután rajzoltatjuk tovább a grafikon. Ha a közös hőmérséklet kialakult, akkor a T megnyomásával új értékek adhatók meg, míg az A megnyomásával előbb itt is megnézhetjük az előző adatokat. A HELP billentyűvel a fenti esetek bármelyikében a program elejéhez, az adatbevitelhez mehetünk vissza.

A program a RUN/STOP billentyűvel állítható le. Ha a leállítás grafikus üzemmódban történt, és így a READY felirat és a villogó

kurzor nem jelent meg, akkor a GRAPHIC 0 "vakon" történő beírásával és a RETURN billentyű megnyomásával térhetünk vissza a normál, szöveges üzemmóddhoz. A funkcióbillentyűk a leállítás után átprogramozottak maradnak. Ez a helyzet a RESET gomb megnyomásával megszüntethető, de ez az eljárás a programot is törli.

```
10 rem melegedes-hules 1.
20 rem zatonyi sandor bekescsaba 1986.
30 fort=1to7:keyt,"":next:key8,chr$(140):graphic1:color0,2,7:color4,2,7:color1,1
40 graphic0,1:printchr$(14):printchr$(8):char1,10,0,"MELEGEDES
- HULES 1."
50 defrne(t)=int(t+.5):defnfnk(t)=(134.5-t)/2:sp$=" " :poke
740,212
60 j=5:k$="m1 = ? ":m$="kg":mi=0:ma=20:gosub350:m1=b
70 j=8:k$="m2 = ? ":gosub350:m2=b
80 c=4.2
90 j=13:k$="T1 = ? ":m$="MoC":mi=-1:ma=100:gosub350:t1=b
100 j=16:k$="T2 = ? ":gosub350:t2=b
110 graphic1,i:box1,31,2,176,160
120 fort=10to15:color0,3,4:char0,28,t,left$(sp$,6):next:circle1,2
48,88,5
130 paint1,248,88:box1,245,80,251,88,,1:color0,2,7:box1,245,2,251
,79
140 fort=10to15:color0,7,5:char0,34,t,left$(sp$,6):next:circle1,2
96,88,5
150 paint1,296,88:box1,293,80,299,88,,1:color0,2,7:box1,293,2,299
,79
160 ift1<t2thent=t1:t1=t2:t2=t:t=m1:mi=m2:m2=t
170 box1,247,fnk(t1),249,78,,1:box1,295,fnk(t2),297,78,,1
180 draw1,240,fnk(t1)to244,fnk(t1):draw1,288,fnk(t2)to292,fnk(t2)
190 fort=12to152step10:draw1,26,tto30,t:draw1,177,ttol81,t:next
200 char1,0,4,"100":char1,1,10,"50":char1,2,16,"0"
210 circle1,253,144,2.5:circle1,309,144,2.5:char1,32,18,"C":char1
,39,18,"C"
220 gosub330:char1,9,10,"START = S":getkeya$:ifa$<"s"anda$<"S"
then220
230 fort=1to19:char1,4,t,sp$+sp$:next
240 fort=32to175:q=.08*(t1-t2):ifq<.04thent1=t2
250 gosub330:box0,247,3,249,fnk(t1),,1:box1,295,79,297,fnk(t2),,1
260 draw1,t,132.5-t1:draw1,t,132.5-t2:t1=t1-q/(c*m1):t2=t2+q/(c*m
2):next
270 iffne(t1)-fne(t2)<1thenchar1,23,(132.5-t1)/8,str$(fne(t1))
280 char1,7,24,"ADATOK = A":char1,23,24,"TOVABB = T"
290 getkeya$:ifa$=chr$(140)thenrun:elseifa$="a"ora$="A"then320
300 ifa$<"t"anda$<"T"then290
```

```

310 iffne(t1)-fne(t2)>=1thengraphic1:char1,7,24,sp$+sp$+sp$:goto2
30:elserun
320 graphic0:char1,23,24,"TOVABB =  T":goto290
330 char,28,18,right$(sp$+str$(fne(t1)),3)
340 char,35,18,right$(sp$+str$(fne(t2)),3):return
350 char1,0,j,sp$+sp$+sp$:char1,13,j,m$:char1,0,j,k$:b$=""
360 print"  T";:getkeya$:ifa$=chr$(140)thenrun:elseifa$=chr$
(13)then400
370 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):a$="  T":goto
390
380 if(a$<"0"ora$>"9")anda$<>" "anda$<>"-"then360:elseb$=b$+a$
390 printa$;:iflen(b$)<4then360
400 print" ";:b=val(b$):ifb<=miorb>mathen350:elsereturn

```

	osztály	csoport	egyéni
új ismeret	●	●	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 85 s (30 mp.)  
 Speciális karakter: nincs

A program két test termikus kölcsönhatásának szimulálására alkalmas. A két test tömege, fajhője és hőmérséklete - bizonyos korlátok közt - szabadon megadható. Az indítás után a többi programtól eltérően - a kis memóriaméret miatt - ismertető nem jelenik meg, hanem azonnal a tömegek, a fajhők, majd a hőmérsékletek értékét kéri a program. A tömegek maximális értéke 20 kg, a fajhőké 20 kJ/(kg°C) lehet. A hőmérsékleteknek -20°C és 120°C közöttieknek kell lenni. Ha a beírt érték négy karakternél rövidebb, a bevitelt a RETURN billentyűvel le kell zárni.

Az adatok beírása után a program sematikus felrajzolja a két, egymással érintkező testet egy-egy hőmérővel, továbbá a hőmérséklet - idő grafikonhoz szükséges rendszert. A képernyőn látható feliratnak megfelelően az S gomb megnyomására (START) folytatódik a program. A gép szimulálja a két test hőmérsékletének változását és a pillanatnyi értékeket kiírja, illetve a hőmérőkön valamint a grafikonon is ábrázolja őket.

Ha kialakult a közös hőmérséklet, akkor ezt a grafikon végéhez is kiírja a program. Ellenkező esetben a T gomb megnyomásával tovább folytathatjuk a rajzolást, vagy előbb az A megnyomásával megnézhetjük a kiindulási adatokat, és csak ezután rajzoltatjuk tovább a grafikon. Ha a közös hőmérséklet kialakult, akkor a T megnyomásával új értékek adhatók meg, míg az A megnyomásával előbb itt is megnézhetjük az előző adatokat. A HELP billentyűvel a fenti esetek bármelyikében a program elejéhez, az adatbevitelhet mehetünk vissza.

A program a RUN/STOP billentyűvel állítható le. Ha a leállítás grafikus üzemmódban történt, és így a READY felirat és a villogó kurzor nem jelent meg, akkor a GRAPHIC 0 "vakon" történő beírásával és a RETURN megnyomásával térhetünk vissza a normál, szöveges üzemmódbhoz. A funkcióbillentyűk a leállítás után átprogramozottak maradnak. Ez a helyzet a RESET gomb megnyomásával megszüntethető, de ez az eljárás a programot is törli.

```

10 a$="MELEGEDES - HULES 2."
20 fort=1to7:keyt,"":next:key8,chr$(140):graphic1:color0,2,7:color1,1:color4,2,7
30 graphic0,1:printchr$(14):printchr$(8):char1,10,0,a$
40 deffne(t)=int(t+.5):deffnk(t)=(134.5-t)/2:sp$="":poke740,212
50 j=5:k$="m1 = ? ":m$="kg":mi=0:ma=20:gosub350:m1=b
60 j=8:k$="m2 = ? ":gosub350:m2=b
70 j=13:k$="c1 = ? ":m$="kJ/(kg*1000)":gosub350:c1=b
80-j=16:k$="c2 = ? ":m$="kJ/(kg*1000)":gosub350:c2=b
90 j=21:k$="T1 = ? ":m$="1000":mi=-21:ma=120:gosub350:t1=b
100 j=24:k$="T2 = ? ":gosub350:t2=b
110 graphic1,1:box1,31,2,176,160
120 fort=10to15:color0,3,4:char0,28,t,left$(sp$,6):next:circle1,248,88,5
130 paint1,248,88:box1,245,80,251,88,,1:color0,2,7:box1,245,2,251,79
140 fort=10to15:color0,7,5:char0,34,t,left$(sp$,6):next:circle1,296,88,5
150 paint1,296,88:box1,293,80,299,88,,1:color0,2,7:box1,293,2,299,79
160 ift1<t2thent=t1:t1=t2:t2=t:t=m1:m1=m2:m2=t:t=c1:c1=c2:c2=t
170 box1,247,fnk(t1),249,78,,1:box1,295,fnk(t2),297,78,,1
180 draw1,240,fnk(t1)to244,fnk(t1):draw1,288,fnk(t2)to292,fnk(t2)
190 fort=12to152step10:draw1,26,tto30,t:draw1,177,ttol81,t:next
200 char1,0,4,"100":char1,1,10,"50":char1,2,16,"0"
210 circle1,253,144,2:circle1,309,144,2:char1,32,18,"C":char1,39,18,"C"
220 gosub330:char1,9,10,"START = S":getkeya$:ifa$><"s"anda$<>"S"then220
230 fort=1to19:char1,4,t,sp$+sp$:next
240 fort=32to175:q=.08*(t1-t2):ifq<.04thent1=t2
250 gosub330:box0,247,3,249,fnk(t1),,1:box1,295,79,297,fnk(t2),,1
260 draw1,t,132.5-t1:draw1,t,132.5-t2:t1=t1-q/(c1*m1):t2=t2+q/(c2*m2):next
270 iffne(t1)-fne(t2)<1thenchar1,23,(132.5-t1)/8,str$(fne(t1))
280 char1,7,24,"ADATOK = A":char1,23,24,"TOVABB = T"
290 getkeya$:ifa$=chr$(140)thenrun:elseifa$="a"ora$="A"then320
300 ifa$<>"t"anda$<>"T"then290
310 iffne(t1)-fne(t2)>=1thengraphic1:char1,7,24,sp$+sp$+sp$:goto230:elserun
320 graphic0:char1,23,24,"TOVABB = T":goto290
330 char,28,18,right$(sp$+str$(fne(t1)),3)
340 char,35,18,right$(sp$+str$(fne(t2)),3):return
350 char1,0,j,sp$+sp$+sp$:char1,13,j,m$:char1,0,j,k$:b$=""
360 print"  ";getkeya$:ifa$=chr$(140)thenrun:elseifa$=chr$(13)then400
370 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):a$="  ":goto390
380 if(a$<"0"ora$>"9")anda$<>". "anda$<>"-"then360:elseb$=b$+a$
390 printa$;:iflen(b$)<4then360
400 print" ";:b=val(b$):ifb<=miorb>mathen350:elsereturn

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 120 s (32 mp.)

Segédprogram: nincs

Speciális karakter: @ → ° £ → Δ

A programmal a szilárd testek hőtágulása tanulmányozható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10 anyagra vonatkozó adatok szerepelnek. Az első oszlopban az anyag neve, a másodikban a kezdeti hosszúság, a harmadikban a hőmérséklet (kezdeti értéke 10°C), az utolsóban pedig a hosszúságváltozás értéke található.

Az első anyag hossza előtt álló > jel a kurzormozgató billentyűvel (nyilakkal) bármelyik anyag hossza, hőmérséklete, vagy hosszváltozása elé vihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban (sötét alapon világos karakterek) jelenik meg a képernyőn.

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Például a ↓ kétszeri és a → egyszeri megnyomása után a harmadik hőmérséklet módosítható. Ezzel az eljárással egy kiválasztott anyag segítségével a hosszváltozás és a másik két mennyiség közti egyenes arányosság egyszerűen tanulmányozható.

Lehetőség van valamennyi anyag valamely adatának egyszerre történő megváltoztatására is. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja.

A programban szereplő anyagok a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni a kívánt anyag nevét, majd az idézőjelen kívülre, a névtől egy vesszővel elválasztva a

hőtágulási tényezőjét. Ügyeljünk arra, hogy a tizedesvessző helyett tizedespontot kell írni. Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A karakterkészlet átírása miatt a képernyő a leállítás után elkövetett esetleges hiba miatt olvashatatlaná válhat. Ez a helyzet az f1 billentyű megnyomásával megszüntethető. Ezzel a módszerrel a program sem törlődik. A gép a program leállítás után is a módosított karakterkészletet használja, továbbá a funkció-billentyűk is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ezzel a program is törlődik.

```
10 rem hotagulas 1.(tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnc1r
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 fori=0to7:reada:poke15584+i,a:next
120 poke65298,56:poke65299,56
130 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
140 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
150 data24,36,36,24,0,0,0,0
160 data8,20,20,36,34,66,126,0
170 key1,"poke65298,56:poke65299,56"+chr$(13)
180 fori=2to7:keyi,"":next
190 key8,chr$(140)
200 color0,2,6:color1,1,0:color4,2,6
210 printchr$(14);chr$(8)
220 print"          HOTAGULAS"
230 print"          1000    £1    (tabl.1.)"
240 print"AAAA programmal 10 kulonbozo szilard anyag"
250 print"hotagulasa tanulmanyozhato."
260 print"AA > jel a billentyuzeten talalhato"
270 print"nyilakkal a megfelelo helyre viheto."
280 print"AA kivant ertek beirasa utan a RETURN"
290 print"gomb megnyomasara a program atirja a"
300 print"tablazat adatait."
310 print"AA beirt ertek legfeljebb 4 karakterbol"
320 print"allhat."
330 print"AAz ESC billentyu megnyomasa utan az"
340 print"egesz oszlop erteke atirhato."
350 char1,25,24,"TOVABB = AT"
360 fori=0to9:readnev$(i),z(i),t(i):next
370 getkeya$:ifa$=chr$(140)thenrun
```

```

380 ifa$<>"t"anda$<>"T"then370
390 color0,10,2:color1,2,7:color4,10,2
400 fori=0to9:x(i)=1:y(i)=10:w(i)=x(i)*y(i)*z(i):next
410 p=1:q=0
420 scnclr
430 print"          1000          £T          £1"
440 print"          (m)          (@C)          (mm)"
450 fori=0to9:gosub630:next
460 b$=""
470 char1,10*p,4+2*q,">"
480 getkeya$:ifa$=chr$(140)thenrun
490 ifa$="|"andp>1andb$=""thenp=p-1:print" ":goto460
500 ifa$="|"andp<3andb$=""thenp=p+1:print" ":goto460
510 ifa$="|"andq>0andb$=""thenq=q-1:print" ":goto460
520 ifa$="|"andq<9andb$=""thenq=q+1:print" ":goto460
530 ifa$="|"andb$=""thenprint" ":p=1:q=0:goto460
540 ifa$="|"andb$=""thenp=1:q=0:fori=0to9:x(i)=1:y(i)=10:gosub630
:next:goto460
550 ifa$=chr$(27)andb$=""thengosub720:goto460
560 if(a$<"0"ora$>"9")anda$<>". "then480
570 i=q
580 char1,10*p,4+2*i," "
590 char1,10*p,4+2*i,"> "
600 gosub810
610 gosub880
620 goto460
630 :
640 ifp<3thenw(i)=x(i)*y(i)*z(i)
650 ifp=3theny(i)=w(i)/(x(i)*z(i)):ify(i)>t(i)theny(i)=1e20:w(i)=
1e20
660 char1,0,4+2*i,nev$(i)
670 char1,10,4+2*i," "
680 printusing"#####. # ";x(i);
690 printusing" ##### ";y(i);
700 printusing" #####.##";w(i);
710 return
720 :
730 print"||| ":char1,10*p,24,"> "
740 gosub810
750 fori=0to9:gosub880:next
760 char1,10*p,24," "
770 q=0
780 return
790 :
800 getkeya$
810 ifa$=chr$(140)thenrun
820 ifa$=chr$(13)andb$>""then870
830 ifa$=chr$(20)andb$>""thenb$=left$(b$,len(b$)-1):print"||| |||";:g
oto800
840 if(a$>"9"ora$<"0")anda$<>". "then800
850 b$=b$+a$
860 print"|||"+a$;" ";:iflen(b$)<4then800
870 return

```



```
880 :
890 b=val(b$)
900 ifp=1thenx(i)=b
910 ifp=2theny(i)=b:ify(i)>=t(i)theny(i)=1e20
920 ifp=3thenw(i)=b
930 gosub630
940 return
950 :
960 data"Wolfram",.0024,3410
970 data"Uveg",.009,700
980 data"Acel",.011,1500
990 data"Vas",.012,1539
1000 data"Beton",.012,700
1010 data"Rez",.017,1083
1020 data"Aluminium",.024,660
1030 data"Cink",.03,419
1040 data"Natrium",.072,97
1050 data"Ken",.09,112
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 120 s (32 mp.)

Speciális karakter: @ → ◊ £ → Δ [ → ≡

A programmal a folyadékok és gázok viselkedése vizsgálható. Elindítás után egy ismertető jelenik meg a képernyőn, amely a program kezelésével kapcsolatos tudnivalókat tartalmazza. Elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő táblázatban 10 anyagra vonatkozó adatok szerepelnek. Az első oszlopban az anyag neve, a másodikban a kezdeti térfogat, a harmadikban a hőmérséklet (kezdeti értéke 10°C), az utolsóban pedig a térfogatváltozás értéke található.

Az első anyag térfogata előtt álló > jel a kurzormozgató billentyűkkel (nyilakkal) bármelyik anyag térfogata, hőmérséklete vagy térfogatváltozása elé vihető. Az így kiválasztott mennyiség értéke módosítható. Ehhez a megfelelő érték beírása után le kell nyomni a RETURN billentyűt. Ha a beírt érték hossza 4 karakter, akkor a gép a RETURN lenyomása nélkül is, automatikusan átírja az adott mennyiséget. A beírás közben az adott érték inverz módban (sötét alapon világos karakterek) jelenik meg a képernyőn.

A program a módosított érték alapján a táblázat megfelelő adatait is azonnal átírja. Például a ↓ kétszeri és a → egyszeri megnyomása után a harmadik hőmérséklet módosítható. Ezzel az eljárással egy kiválasztott anyag segítségével a térfogatváltozás és a másik két mennyiség közti egyes arányosság egyszerűen tanulmányozható.

Lehetőség van valamennyi anyag adatának egyszerre történő megváltoztatására is. Ehhez először az ESC gombot kell lenyomni. Ennek hatására a > jel a képernyő legalsó, üres sorába ugrik. A megfelelő érték beírása után a program az adott oszlop minden egyes sorába beírja ezt az értéket, a táblázat többi adatát pedig ennek megfelelően módosítja.

A programban szereplő anyagok a lista végén található sorok átírásával egyszerűen megváltoztathatók. Ehhez a DATA utasítás után idézőjelek között be kell írni a kívánt anyag nevét, az idézőjelen kívülre, a névtől egy vesszővel elválasztva a hőtágulási tényezőjét, majd újabb vessző után a forráspont értékét illetve gázoknál az 1e4 értéket. Ügyeljünk arra, hogy a tizedes-

vessző helyett tizedespontot kell írni. Az átírt sort a RETURN billentyű megnyomásával le kell zárni.

A program futása a RUN/STOP billentyűvel állítható meg. A karakterkészlet átírása miatt a képernyő a leállítás után elkövetett esetleges hiba miatt olvashatatlanul válhat. Ez a helyzet az f1 billentyű megnyomásával megszüntethető. Ezzel a módszerrel a program sem törlődik. A gép a program leállítása után is a módosított karakterkészletet használja, továbbá a funkció-billentyűk is átprogramozott állapotban maradnak. Ez a RESET gomb megnyomásával megszüntethető, de ezzel a program is törlődik.

```
10 rem hotagulas 2.(tabl.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 fori=0to7:reada:poke15584+i,a:next
120 fori=0to7:reada:poke15576+i,a:next
130 poke65298,56:poke65299,56
140 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
150 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
160 data24,36,36,24,0,0,0,0
170 data8,20,20,36,34,66,126,0
180 data56,4,56,4,56,0,0,0
190 key1,"poke65298,56:poke65299,56"+chr$(13)
200 fori=2to7:keyi,"":next
210 key8,chr$(140)
220 color0,2,7:color4,2,7
230 printchr$(14);chr$(8)
240 print"          HOTAGULAS"
250 print"          o+++++++||| "
260 print"          (tabl.1.)"
270 print"AAAAA programmal 10 kulonbozo folyadek es"
280 print"gaz hotagulasa tanulmanyozhato allando"
290 print"0.1 MPa nyomason. (T0= 20 @C)"
300 print" A > jel a billentyuzeten talalhato"
310 print"nyilakkal a megfelelo helyre viheto."
320 print" A kivant ertekek beirasa utan a RETURN"
330 print"gomb megnyomasara a program atirja a"
340 print"tablazat adatait."
350 print" A beirt ertekek legfeljebb 4 karakterbol"
360 print"allhat."
370 print" A Az ESC billentyu megnyomasa utan az"
380 print"egesz oszlop erteke atirhato."
```

```

390 char1,25,24,"TOVABB =  T"
400 fori=0to9:readnev$(i),z(i),t(i):next
410 getkeya$:ifa$=chr$(140)thenrun
420 ifa$<>"t"anda$<>"T"then410
430 color0,10,2:color1,2,7:color4,10,2
440 fori=0to9:x(i)=1:y(i)=10:w(i)=x(i)*y(i)*z(i)/1e3:next
450 p=1:q=0
460 scncclr
470 print"          V    £T          £V"
480 print"          (dm[)    (@C)    (dm[)"
490 fori=0to9:gosub670:next
500 b$=""
510 char1,10*p,4+2*q,">"
520 getkeya$:ifa$=chr$(140)thenrun
530 ifa$=" "andp>1andb$=""thenp=p-1:print" ":goto500
540 ifa$=" "andp<3andb$=""thenp=p+1:print" ":goto500
550 ifa$=" "andq>0andb$=""thenq=q-1:print" ":goto500
560 ifa$=" "andq<9andb$=""thenq=q+1:print" ":goto500
570 ifa$=" "andb$=""thenprint" ":p=1:q=0:goto500
580 ifa$=" "andb$=""thenp=1:q=0:fori=0to9:x(i)=1:y(i)=10:gosub670
:next:goto500
590 ifa$=chr$(27)andb$=""thengosub760:goto500
600 if(a$<"0"ora$>"9")anda$<>". "then520
610 i=q
620 char1,10*p,4+2*i," "
630 char1,10*p,4+2*i,"> "
640 gosub850
650 gosub920
660 goto500
670 :
680 ifp<3thenw(i)=x(i)*y(i)*z(i)/1e3
690 ifp=3theny(i)=w(i)*1e3/(x(i)*z(i)):ify(i)>t(i)-20theny(i)=1e2
0:w(i)=1e20
700 char1,0,4+2*i,nev$(i)
710 char1,10,4+2*i," "
720 printusing"#####.# ";x(i);
730 printusing" ##### ";y(i);
740 printusing"###.### ";w(i);
750 return
760 :
770 print" " :char1,10*p,24,"> "
780 gosub850
790 fori=0to9:gosub920:next
800 char1,10*p,24," "
810 q=0
820 return
830 :
840 getkeya$
850 ifa$=chr$(140)thenrun
860 ifa$=chr$(13)andb$>" "then910

```

```

870 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1):print"  ";g
oto840
880 if(a$>"9"ora$<"0")anda$<>". "then840
890 b$=b$+a$
900 print" "+a$;" ";:iflen(b$)<4then840
910 return
920 :
930 b=val(b$)
940 ifp=1thenx(i)=b
950 ifp=2theny(i)=b:ify(i)>=t(i)-20theny(i)=1e20
960 ifp=3thenw(i)=b
970 gosub670
980 return
990 :
1000 data"Viz",.13,100
1010 data"Higany",.18,357
1020 data"Kensav",.56,338
1030 data"Benzin",1,80
1040 data"Alkohol",1.1,78
1050 data"Eter",1.62,34
1060 data"Hidrogen",3.4,1e4
1070 data"Nitrogen",3.4,1e4
1080 data"Oxigen",3.4,1e4
1090 data"Levego",3.4,1e4

```

## 34. FÉNYTERJEDEÉS

8. osztály

	osztály	csoport	egyéni
új ismeret	●	●	●
alkalmazás	○	○	○
ellenőrzés			

Betöltési idő: 80 s (29 mp.)

Speciális karakter: nincs

A programmal a fény viselkedése szimulálható két átlátszó közeg határfelületén. Az indítás után megjelenő ismertető elolvasása után ki kell választani a megadott 4 lehetőség közül, hogy milyen anyagok határán akarjuk a fény viselkedését vizsgálni. (levegő - víz, levegő - üveg, levegő - gyémánt és víz - gyémánt.) A választáshoz csak a megfelelő betűt kell lenyomni.

A program ezek után kis ideig számol, majd felrajzolja a két közeg határfelületét, a beesési merőleget és egy, a felületre merőlegesen beeső fénysugár útját, valamint kiírja a beesési, visszaverődési és törési szög pillanatnyi értékét. A program a valóságos helyzetnek megfelelően egyszerre ábrázolja a visszavert és a megtört fénysugarat, így a program mindkét jelenség tanításához használható.

A  $\uparrow$  és  $\downarrow$  gombokkal fokenként, a + és - gombokkal 10 fokenként változtatható a beesési szög nagysága. Az ennek megfelelő szögértékeket a program kiírja a képernyőre és azonnal berajzolja a megfelelő fénysugarakat is. (A beesési szög értéke legfeljebb 89 fok lehet.) Ha a  $\uparrow$  gombot folyamatosan lenyomva tartjuk, jól látszik, hogy a beesési szög és a törési szög nem egyenesen arányos egymással, és könnyen eljuthatunk a határszög fogalmának megsejtetéséig is.

A HELP megnyomásával itt is az ismertetőhöz juthatunk vissza, és ekkor lehetőség nyílik a két közeg megváltoztatására.

A program futása a RUN/STOP billentyűvel állítható meg. Ha a leállítás grafikus üzemmódban történt, és ezért a READY üzenet illetve a villogó kurzor nem jelent meg, a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN lenyomásával visszatérhetünk a normál szöveges üzemmódkhoz. A gép funkcióbillentyűi a leállítás után is átprogramozott állapotban maradnak.

Ez a helyzet a RESET gomb megnyomásával megszüntethető, de ez a programot is törli a memóriából.

```
10 rem fenyterjedes
20 rem zatonyi sandor bekescsaba 1986.
30 graphic0,1:printchr$(14);chr$(8);:key8,chr$(8):poke740,212
40 color0,7,0:color4,7,0:color1,2
50 forc=1to7:keyc,"":next
60 char1,14,0,"FENYTERJEDES":print
70 print"███A programmal a feny ket kozeg hataran"
80 print"torteno viselkedese tanulmanyozhato."
90 print"██A beesesi szog a fel- es lefele mutato"
100 print"nyilakkal fokonkent, a + es - gombokkal 10 fokonkent va
ltoztathato."
110 print"██A megfelelo betu lenyomasaval lehet a ket kozeket ki
valasztani."
120 char1,9,18,"██A. LEVEGO → █VIZ"
130 char1,9,20,"██B. LEVEGO → █UVEG"
140 char1,9,22,"██C. LEVEGO → █GYEMANT"
150 char1,9,24,"██D. █VIZ → █GYEMANT"
160 getkeya$:a=asc(a$)and127:s1=2
170 ifa=65thenn=1.33:s2=16:goto210
180 ifa=66thenn=1.5:s2=14:goto210
190 ifa=67thenn=2.42:s2=5:goto210
200 ifa=68thenn=1.82:s1=16:s2=5:else160
210 color0,s1,7:color1,1
220 graphic1,1:k=.75:dimbe%(89):char,20,9,"dolgozom"
230 foral=1to89:x=sin(al*%180)/n:be%(al)=atn(x/sqr(1-x*x))*180/%
:next
240 sp$=" "
250 color0,s2,6:color1,1
260 fori=0to24:char1,20,i,sp$:next
270 char1,20,1,"beesesi szog"
280 char1,20,5,"visszaver. szog"
290 char1,20,9,"toresi szog"
300 forc=0to8step4:char1,38,c,"o":next:al=0
310 forc=12036to12355step8:pokec,51:next
320 be=be%(al)
330 j=1:gosub420:color0,s2,6:char1,20,22," "
340 char1,35,1,str$(al)+" ":char1,35,5,str$(al)+" ":char1,35,9,st
r$(be)+" "
350 getkeya$
360 ifa$=chr$(8)thenrun
370 ifa$="█"andal<89thenj=0:gosub420:al=al+1:goto310
380 ifa$="+"andal<80thenj=0:gosub420:al=al+10:goto310
390 ifa$="█"andal>0thenj=0:gosub420:al=al-1:goto310
400 ifa$="-"andal>9thenj=0:gosub420:al=al-10:goto310
410 goto350
420 color0,s1,7
430 drawj,159,100to200;270+al:drawj,159,100to150-al*k;270-al
440 circlej,,2,6,,270-al,120:color0,s2,6
450 drawj,160,100to150-be*k;90+be
460 circlej,,2,6,,90+be,120:return
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 80 s (30 mp.)

Speciális karakter: nincs

A program a tükrök képalkotásának vizsgálatához készült. Indítás után egy rövid ismertető jelenik meg a képernyőn. Innen a tükör fókusztávolságának megadásával léphetünk tovább. A kívánt érték beírását a RETURN billentyű megnyomásával kell lezárni. A pozitív érték homorú tükört, a negatív domború tükört határoz meg. A beírt érték abszolút értéke 10 és 60 közti lehet.

A képernyőn ezután kirajzolódik a tükör optikai tengelye, fősíkja és a program az optikai tengelyen bejelöli a fókusztávolságnak és a kétszeres fókusztávolságnak megfelelő pontokat. Ezt követően a program kiírja, hogy milyen típusú tükörről van szó és mekkora a fókusztávolság. Induló adatnak 100 cm tárgytávolságot választva feltünteti a tárgy- és képtávolságot, és ennek megfelelően berajzolja a tárgyat, illetve a képet jelképező két szakaszt a megfelelő helyre.

Ezekután a billentyűzeten található négy nyíllal megváltoztathatjuk a tárgy helyzetét, illetve nagyságát. A program ennek megfelelően meghatározza és kiírja a tárgytávolság és a képtávolság értékét és ezek alapján ábrázolja a tárgyat, illetve a képet.

Mód van arra is, hogy az 1 . . . 4 billentyűk megnyomásával az adott helyzetben egy-egy nevezetes fénysugarat megjelenítsünk. Az 5 gomb megnyomásával a program a négy nevezetes sugarat egyszerre rajzolja meg.

A CLEAR/HOME billentyű megnyomásával a program visszaáll a 100 cm-es tárgytávolságnak megfelelő kiindulási helyzetbe. A SHIFT és a CLEAR/HOME együttes megnyomásával vagy a HELP leütésével a bevezetőhöz térhetünk vissza, és új fókusztávolságot adhatunk meg. A program a RUN/STOP billentyűvel állítható meg. Ha ez grafikus üzemmódban történt, és ezért a READY üzenet valamint a villogó kurzor nem jelent meg, a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN lenyomásával visszatérhetünk a szöveges üzemmódba. A gép funkcióbillentyűi a leállítás után is



átprogramozva maradnak. Ez az állapot a RESET megnyomásával megszüntethető, de ez a programot is törli a memóriából.

```
10 rem tukrok
20 rem zatonyi sandor bekescsaba 1986.
30 graphic0,1:printchr$(14);chr$(8):key8,chr$(140):poke740,212
40 forc=1to7:keyc,"":next:color0,7,1:color4,7,1:color1,2,7
50 print" 1 1 1"
60 print" - + - = - TUKROK"
70 print" t k f"
80 print"███A tukur fokusztaavolsaga 10 es 60 cm koztlehet."
90 print"███A tárgy helyzete es nagysaga a nyil gombokkal változtatható."
100 print"███Az 1-5 gombokkal a nevezetes sugarak rajzoltathatok meg."
110 print"███Mekkora a fokusztaavolsag (cm) ";:f$=""
120 print" ←███";:getkeya$
130 ifa$=chr$(13)andf$>" "then180
140 ifa$=chr$(20)andf$>" "thenf$=left$(f$,len(f$)-1):goto170
150 ifa$="-"andf$=""thenf$=f$+a$:goto170
160 ifa$<"0"ora$>"9"then120:elseif$=f$+a$
170 printa$;:iflen(f$)<4then120
180 f=int(10*val(f$)+.5)/10:af=abs(f):sf=5*sgn(f)
190 l$="Homofu":iff<0thenl$="Domboru"
200 ifaf<10oraf>60thenrun
210 graphic1,1
220 tt=100:tn=50:x0=160:y0=100:gosub300
230 gosub380
240 gosub430:getkeyc$:gosub450:gosub340
250 ifc$="█"thentt=tt+1:goto230:elseifc$=""andtt>1thentt=tt-1:goto230
260 ifc$="█"andtn<66thentn=tn+5:goto230:elseifc$=""andtn>5thentn=tn-5:goto230
270 ifc$="█"orc$=chr$(140)thengraphic0:run:elseifc$=""then210
280 iftt<>fthenonval(c$)goto460,470,480,490,500
290 goto240
300 scnc1r:draw1,0,y0to320,y0:draw1,x0,10tox0,190
310 draw1,160-sf,5tox0,10:draw1,160-sf,195tox0,190
320 char1,26,22,"f =" +str$(f)+" cm"
330 char1,26,24,l$+" tukur"
340 draw1,x0-af,y0tox0-af,95:draw1,x0-2*af,y0tox0-2*af,95
350 draw1,x0+af,y0tox0+af,105:draw1,x0+2*af,y0tox0+2*af,105
360 char1,20+af/8,14,"F":char1,20-af/8,11,"F"
370 return
380 iftt=fthenchar1,3,2," nincs ":goto410
390 kt=int(10/(1/f-1/tt)+.5)/10:kn=tn*kt/tt:kt$=str$(kt)
400 char1,0,2,"k =" +kt$+" cm "
```

```
410 x1=x0-tt:x2=x0-kt:y1=y0-tn:y2=y0+kn
420 char1,0,0,"t="+str$(tt)+" cm ":return
430 draw1,x1,99tox1,y1:iftt<>fthendraw1,x2,y0tox2,y2
440 return
450 draw0,x1,99tox1,y1:draw0,x2,y0tox2,y2:draw1,x2,y0:return
460 draw1,x1,y1tox0,y1tox2,y2:goto510
470 draw1,x1,y1tox0,y2tox2,y2:goto510
480 draw1,x1,y1tox0,y0tox2,y2:goto510
490 iftt=2*fthen510:elsedraw1,x1,y1tox0,y0+2*tn*f/(tt-2*f)tox2,y2
:goto510
500 draw1,x1,y1tox0,y1tox2,y2tox0,y2tox1,y1tox0,y0tox2,y2:goto490
510 gosub430:getkeyc$:gosub300:gosub380:goto250
```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás	○	○	●
ellenőrzés			

Betöltési idő: 85 s (30 mp.)

Speciális karakter: nincs

A program a lencsék képalkotásának vizsgálatához készült. Indítás után egy rövid ismertető jelenik meg a képernyőn. Innen a lencse fókusztávolságának megadásával léphetünk tovább. A kívánt érték beírását a RETURN billentyű megnyomásával kell lezárni. A pozitív érték gyűjtőlencsét, a negatív szórólencsét határoz meg. A beírt érték abszolút értéke 10 és 60 közti lehet.

A képernyőn ezután kirajzolódik a lencse optikai tengelye, fősíkja és a program az optikai tengelyen bejelöli a fókusztávolságnak és a kétszeres fókusztávolságnak megfelelő pontokat. Ezt követően a program kiírja, hogy milyen típusú lencséről van szó és mekkora a fókusztávolság. Induló adatnak 100 cm tárgytávolságot választva feltünteti a tárgy- és képtávolságot, és ennek megfelelően berajzolja a tárgyat, illetve a képet jelképező két szakaszt a megfelelő helyre.

Ezekután a billentyűzeten található négy nyíllal megváltoztathatjuk a tárgy helyzetét, illetve nagyságát. A program ennek megfelelően újra és újra meghatározza és kiírja a tárgytávolság és a képtávolság értékét, és ezek alapján ábrázolja a tárgyat, illetve a képet.

Mód van arra is, hogy az 1 . . . 4 billentyűk megnyomásával az adott helyzetben egy-egy nevezetes fénysugarat megjelenítsünk. Az 5 gomb megnyomásával a program a négy nevezetes sugarat egyszerre rajzolja meg.

A CLEAR/HOME billentyű megnyomásával a program visszaáll a 100 cm-es tárgytávolságnak megfelelő kiindulási helyzetbe. A SHIFT és a CLEAR/HOME együttes megnyomásával vagy a HELP leütésével a bevezetőhöz térhetünk vissza, és új fókusztávolságot adhatunk meg. A program a RUN/STOP billentyűvel állítható meg. Ha ez grafikus üzemmódban történt, és ezért a READY üzenet valamint a villogó kurzor nem jelent meg, a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN lenyomásával visszatérhetünk a szöveges üzemmódba. A gép funkcióbillentyűi a leállítás után is

átprogramozva maradnak. Ez az állapot a RESET megnyomásával megszüntethető, de ez a programot is törli a memóriából.

```
10 rem lencsek
20 rem zatonyi sandor bekescsaba 1986.
30 graphic0,1:printchr$(14);chr$(8):key8,chr$(140):poke740,212
40 forc=1to7:keyc,"":next:color0,7,2:color4,7,2:color1,2,7
50 print"  1  '1  1"
60 print"  - + - = -          LENCSEK"
70 print"  t  k  f"
80 print"███A lencse fókusztaolsaga 10 es 60 cm      kozt lehet."
90 print"███A tárgy helyzete es nagysaga a nyil      gombokkal válto
ztathato."
100 print"███Az 1-5 gombokkal a nevezetes sugarak      rajzoltathatok
meg."
110 print"███Mekkora a fókusztaolsag (cm) ";:f$=""
120 print"  ←████";:getkeya$
130 ifa$=chr$(13)andf$>" "then180
140 ifa$=chr$(20)andf$>" "thenf$=left$(f$,len(f$)-1):gotol170
150 ifa$="-"andf$=""thenf$="-":gotol170
160 ifa$<"0"oraf$>"9"then120:elseif$=f$+a$
170 printa$;:iflen(f$)<5then120
180 f=int(10*val(f$)+.5)/10:af=abs(f):sf=5*sgn(f)
190 l$="Gyujto":iff<0thenl$="Szoro"
200 ifaf<10oraf>60thenrun
210 graphic1,1
220 tt=100:tn=50:x0=160:y0=100:gosub300
230 gosub370
240 gosub420:getkeyc$:gosub440
250 ifc$="█"thentt=tt+1:goto230:elseifc$=""andtt>1thentt=tt-1:got
o230
260 ifc$="█"andtn<66thentn=tn+5:goto230:elseifc$=""andtn>5thentn=
tn-5:goto230
270 ifc$="█"orc$=chr$(140)thengraphic0:run:elseifc$=""then210
280 iftt<>fthenonval(c$)goto450,460,470,480,490
290 goto240
300 scnc1r:draw1,0,y0to320,y0:draw1,x0,10tox0,190
310 draw1,155,10+sftox0,10tol65,10+sf:draw1,155,190-sftox0,190tol
65,190-sf
320 draw1,x0-af,y0tox0-af,105:draw1,x0-2*af;y0tox0-2*af,105
330 draw1,x0+af,y0tox0+af,95:draw1,x0+2*af,y0tox0+2*af,95
340 char1,20-af/8,14,"F":char1,20+af/8,11,"F"
350 char1,26,1,"f =" +str$(f)+" cm"
360 char1,26,3,l$+" lencse":return
370 iftt=fthenchar1,3,23," nincs ":goto400
380 kt=1/(1/f-1/tt):kn=tn*kt/tt:kt$=str$(int(kt*10+.5)/10)
390 char1,0,23,"k =" +kt$+" cm "
400 x1=x0-tt:x2=x0+kt:y1=y0-tn:y2=y0+kn
```

```
410 char1,0,21,"t="+str$(tt)+" cm ":return
420 draw1,x1,99tox1,y1:if tt<>f then draw1,x2,y0tox2,y2
430 return
440 draw0,x1,99tox1,y1:draw0,x2,y0tox2,y2:draw1,x2,y0:return
450 draw1,x1,y1tox0,y1tox2,y2:goto500
460 draw1,x1,y1tox0,y2tox2,y2:goto500
470 draw1,x1,y1tox0,y0tox2,y2:goto500
480 if tt=2*f then 500:else draw1,x1,y1tox0,y0+2*tn*f/(tt-2*f)tox2,y2
:goto500
490 draw1,x1,y1tox0,y1tox2,y2tox0,y2tox1,y1tox0,y0tox2,y2:goto480
500 gosub420:getkeyc$:gosub300:gosub370:goto250
```

	osztály	csoport	egyéni
új ismeret	●	●	●
alkalmazás			
ellenőrzés			

Betöltési idő: 60 s (28 mp.)

Speciális karakter: nincs

A program az additív színkeverés bemutatására készült. Csak színes televízióval vagy monitorral használható a program! Indítás után egy rövid tájékoztató jelenik meg, melynek elolvasása után a megadott két színösszeállítás egyikét ki kell választani. A választáshoz az 1, illetve 2 gombok egyikét kell csak lenyomni. Ezt követően a program a három kiválasztott szín mindegyikével egy-egy vízszintes vonalából álló téglalapot rajzol. A három téglalap a + gomb többszöri megnyomásával fokozatosan fedésbe hozható egymással. A - megnyomásával újra szétválaszthatjuk őket. Közben megfigyelhetjük, hogy milyen színek alakulnak ki ott ahol kettő, illetve három téglalap "átfedi" egymást. (Igazi átfedés nem jön létre, csak a különböző színű vonalak csúsznak egymás közé, és ez már 1-2 méterről nézve is a színek additív összeadását eredményezi.) Jól beállított televízió esetén középen fehér folt jelenik meg. (Érdeemes a szintelítettséget szabályozó gombot a maximális telítettséghez állítani.)

A CLEAR/HOME hatására a három téglalap visszaáll a kiindulási helyzetbe. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomására a téglalapok szintén visszaállnak az eredeti helyzetükbe, de a másik színekészlet színeit veszik fel. A HELP gomb megnyomásával a bevezetőhöz, és ezzel együtt a színválasztáshoz tér vissza a program.

A program futása a RUN/STOP billentyűvel szakítható meg. Ha a megszakításkor a READY üzenet és a villogó kurzor nem jelenik meg, mert a megszakítás grafikus üzemmódban történt, akkor a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN billentyű lenyomásával térhetünk vissza a normál, szöveges üzemmódbhoz. A funkcióbillentyűk átprogramozott állapota a RESET gomb megnyomásával szüntethető meg, de ez a program törlésével jár együtt.

```

10 rem szinek
20 rem zatonyi sandor bekescsaba 1986.
30 graphic1,1:graphic0,1:color0,1:color1,2,7:color4,1
40 fori=1to7:keyi,"":next:key8,chr$(8):printchr$(14);chr$(8)
50 s1(0)=3:s2(0)=6:s3(0)=7
60 s1(1)=5:s2(1)=4:s3(1)=8
70 char1,23,1,"S Z I N E K █"
80 fori=1to3:forj=1to16:color1,j,(j-1)/2:print"█ █";:next:print""
:next
90 print"███A program segitsegevel a szinek ossze- tetele szemle
ltetheto."
100 print"█A megjeleno harom kulonbozo szinu"
110 print"teglalap a + gombbal kozelitheto, a - gombbal tavolit
hato."
120 print"██Melyik harom szint vizsgaljuk?"
130 print"███ 1. █ █ voros █ █ zold █ █ █ kek"
140 print"███ 2. █ █ bibor █ █ kekeszold █ █ sarga"
150 print"███A szam lenyomasaval lehet valasztani.";
160 getkeya$:ifa$<>"1"anda$<>"2"then160
170 sz=asc(a$)-49
180 color0,1:color4,1
190 graphic3,1
200 color1,s1(sz),5
210 fori=18+yto93+ystep3
220 draw1,7.5+x,ito77.5+x,i
230 next
240 color2,s2(sz),5
250 fori=19+yto94+ystep3
260 draw2,82.5-x,itol152.5-x,i
270 next
280 color3,s3(sz),5
290 fori=107-yto182-ystep3
300 draw3,45,itol15,i
310 next
320 getkeya$
330 ifa$="-"andx>7thenx=x-7.5:y=y-9:goto190
340 ifa$="+"andx<36thenx=x+7.5:y=y+9:goto190
350 ifa$="█"thenx=0:y=0:goto190
360 ifa$="█"thensz=sz+1and1:x=0:y=0:goto190
370 ifa$=chr$(8)thenrun
380 goto320

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○		●
ellenőrzés			●

Betöltési idő: 80 s (29 mp.)

Speciális karakter: nincs

A program az iskolai tanuló kísérleti műszer leolvasásának gyakorlásához használható. Ez a műszer két változatban kétféle skálával került az iskolákba, ezért a programot mindkét változathoz elkészítettük. Az első változat a 6-os beosztású skálával rendelkező műszerekhez, a második pedig az 5-ös skála-beosztású műszerekhez készült.

A program elindítása után egy menü segítségével választhatjuk ki a mérés határt. Ismertetőt a kis memóriaméret miatt nem ad a program. A választáskor csupán a megfelelő betűt kell lenyomni a billentyűzeten. Ezután a megjelenő feliratnak megfelelően választhatunk a gyakorlás és a teszt között. Itt szintén csak a G és a T billentyűk egyikét kell lenyomni. Ennek hatására a program mindkét esetben először felrajzolja a műszer skáláját és a képernyő alján két nyíllal jelzi, hogy mely csatlakozóhoz kötöttük a vezetékeket, azaz milyen mérés határt állítottunk be.

A gyakorlás üzemmódban ezt követően a ↓ és ↑ gombok megnyomásával szabályozhatjuk a mutató kitérését. A közepén megjelenő fekete téglalapban ezzel egyidejűleg számjegyekkel kiírva, (digitális alakban) is megjelenik a megfelelő érték. A CLEAR/HOME billentyű hatására a mérés határ kiválasztáshoz mehetünk vissza. Ugyanezt eredményezi a SHIFT és a CLEAR/HOME együttes lenyomása vagy a HELP megnyomása is, de ekkor az esetleges korábbi teszt eredmények is törlődnek.

A teszt üzemmódban a mutató egy véletlenszerűen megválasztott helyzetbe áll be, és a tanulónak a mérés határ figyelembevételével, a billentyűzet segítségével kell beírni a műszer által mutatott értéket. A beírásakor a mértékegységet is meg kell adni. A beírt adatot, ha az 6 karakternél rövidebb, a RETURN gomb megnyomásával kell lezárni. Beírás közben a beírt karakterek a



```

10 rem v-a muszer 1
20 rem zatonyi sandor bekescsaba 1986.
30 graphic0,1:color0,2,7:color1,1:color4,7,3:printchr$(14);chr$(8
)
40 fori=1to7:keyi,"":next:key8,chr$(8):mh$="30 6.6 3":m$=" 311273
4"
50 char1,9,2,"MERESHATAR KIVALASZTAS"
60 char1,15,6,"A. 30 V":char1,15,9,"B. 6 V"
70 char1,15,12,"C. 0.6 A":char1,15,15,"D. 3 A"
80 char1,3,19,"A megfelelo betu lenyomasaval lehet valasztani
."
90 getkeya$:ifa$<"a"ora$>"d"then90
100 m=asc(a$)-64:me$="v":ifm>2thenme$="a"
110 m=2*m-1:mh=val(mid$(mh$,m,2)):m=val(mid$(m$,m,2))
120 char1,3,23,"Gyakorlas vagy teszt? (G/T)"
130 getkeyt$:ift$=chr$(8)then30
140 ift$="g"then150:elseift$="t"then200:elseif30
150 gosub300:x=0
160 i=1:gosub390:getkeya$:i=0:gosub410
170 ifa$=" "ora$=chr$(8)then30:elseifa$=""thenrun
180 ifa$=" "andx<60thenx=x+1:elseifa$=""andx>0thenx=x-1
190 goto160
200 gosub300:ifn>0thengosub420
210 x=int(60*rnd(0)+1):i=1:gosub410:b$=""
220 getkeya$:ifa$=chr$(8)thengosub390:goto290:elseifa$=" "thenrun
230 ifa$=" "then30:elseifa$=chr$(13)andb$>" "then270
240 ifa$="a"ora$="v"ora$="."or(a$>="0"anda$<="9")thenb$=b$+a$
250 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1)
260 char1,17,15,b$+" ",1:iflen(b$)<6then220
270 dx=abs(val(b$)-mh*x/60)
275 ifdx<.0001andright$(b$,1)=me$thenj=j+1:b$="remek ":elseb$="ro
ssz "
280 n=n+1:char1,17,15,b$,1:gosub420
290 fori=0to1000:next:char1,17,15," ",1:i=0:gosub410:goto210
300 m$=" 30v 6v + 0.6a 3a":k=330
310 graphic1,1:l=10:h=20:gosub350:l=5:h=12:gosub350:l=1:h=5:gosub
350
320 circle1,160,310,215,,330,30:draw1,0,125to52,125:draw1,268,125
to320,125
330 box1,110,112,210,133,,1:char1,0,18,m$
340 char1,19,20,"↑":char1,m,20,"↑":return
350 fori=0to60stepl;ifh=20thenchar1,0,0,str$(i):sshapea$,9,0,14,6
:char1,1,0," "
360 draw0,160,310to270;k+i:draw1toh;k+i
370 ifh=20thenlocatel0;k+i:locate-4,-4:gshapea$
380 next:return
390 x$=str$(mh*x/60)+me$:ifval(x$)<1andx>0thenx$=" 0"+right$(x$,1
en(x$)-1)
400 char1,17,15,x$+" ",1
410 locatel160,310:drawi,216;k+xto50;k+x:return
420 char1,16,23,"jo:"+str$(int(j/n*100))+"% ":return

```

```

10 rem v-a muszer 2
20 rem zatonyi sandor bekescsaba 1986.
30 graphic0,1:color0,2,7:color1,1:color4,8,6:printchr$(14);chr$(8
)
40 fori=1to7:keyi,"":next:key8,chr$(8):mh$="5010 1 5":m$=" 311273
3"
50 char1,9,2,"MERESHATAR KIVALASZTAS"
60 char1,15,6,"A. 25 V":char1,15,9,"B. 5 V"
70 char1,15,12,"C. 0.5 A":char1,15,15,"D. 2.5 A"
80 char1,3,19,"A megfelelo betu lenyomasaval lehet valasztani
."
90 getkeya$:ifa$<"a"ora$>"d"then90
100 m=asc(a$)-64:me$="v":ifm>2thenme$="a"
110 m=2*m-1:mh=val(mid$(mh$,m,2))/2:m=val(mid$(m$,m,2))
120 char1,3,23,"Gyakorlas vagy teszt? (G/T)"
130 getkeyt$:ift$=chr$(8)then30
140 ift$="g"then150:elseift$="t"then200:elseif30
150 gosub310:x=0
160 i=1:gosub400:getkeya$:i=0:gosub420
170 ifa$=" "ora$=chr$(8)then30:elseifa$=""thenrun
180 ifa$=" "andx<50thenx=x+1:elseifa$=""andx>0thenx=x-1
190 goto160
200 gosub310:ifn>0thengosub430
210 x=int(50*rnd(0)+1):i=1:gosub420:b$=""
220 getkeya$:ifa$=chr$(8)thengosub400:goto300:elseifa$=" "thenrun
230 ifa$=" "then30:elseifa$=chr$(13)andb$>" "then270
240 ifa$="a"ora$="v"ora$="."or(a$)="0"anda$<="9")thenb$=b$+a$
250 ifa$=chr$(20)andb$>" "thenb$=left$(b$,len(b$)-1)
260 char1,17,15,b$+" ",1:iflen(b$)<6then220
270 dx=abs(val(b$)-mh*x/50)
280 ifdx<.0001andright$(b$,1)=me$thenj=j+1:b$="remek ":elseb$="ro
ssz "
290 n=n+1:char1,17,15,b$,1:gosub430
300 fori=0to1000:next:char1,17,15," ",1:i=0:gosub420:goto210
310 m$=" 25v 5v + 0.5a 2.5a":k=335
320 graphic1,1:l=10:h=20:gosub360:l=5:h=12:gosub360:l=1:h=5:gosub
360
330 circle1,160,310,215,,330,30:draw1,0,125to52,125:draw1,268,125
to320,125
340 box1,110,112,210,133,,1:char1,0,18,m$
350 char1,19,20,"↑":char1,m,20,"↑":return
360 fori=0to50stepl:ifh=20thenchar1,0,0,str$(i):sshapea$,9,0,14,6
:char1,1,0," "
370 draw0,160,310to270;k+i:draw1toh;k+i
380 ifh=20thenlocatel0;k+i:locate-4,-4:gshapea$
390 next:return
400 x$=str$(mh*x/50)+me$:ifval(x$)<1andx>0thenx$=" 0"+right$(x$,l
en(x$)-1)
410 char1,17,15,x$+" ",1
420 locatel160,310:drawi,216;k+xto50;k+x:return
430 char1,16,23,"jo:"+str$(int(j/n*100))+"% ":return

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés	○	○	●

Betöltési idő: 85 s (30 mp.)

Speciális karakter: nincs

A program Ohm törvényének, illetve az ellenállás fogalmának tanításához alkalmazható. Indítása után egy tájékoztató jelenik meg a képernyőn. Innen a T gomb megnyomásával léphetünk tovább. Ekkor a program felrajzolja a 3. ábrán látható kapcsolási rajzot és egy koordinátarendszert. A fogyasztó ellenállását a gép véletlenszerűen választja meg. A ↑ és a ↓ gombokkal a feszültség voltonként növelhető, illetve csökkenthető. (A feszültség kezdeti értéke 0 V.)

A fogyasztón átfolyó áram erősségét a program kiírja és ezzel egyidőben az aktuális értéknek megfelelő pontot berajzolja az áramerősség - feszültség grafikonba.

Az R billentyű megnyomására a program kiírja az ellenállás értékét, így a feszültségből és áramerősségből számolt érték azonnal ellenőrizhető. Ha a feszültség a 150 V-os vagy az áramerősség a 15 A-es határt eléri, a gép a grafikon végéhez kiírja az ellenállás értékét, és véletlenszerűen új ellenállást ad.

Ugyanez történik, ha az U gombot nyomjuk meg. Az új ellenállás az előzőekben leírtak szerint vizsgálható. A grafikont a program ugyanabban a koordinátarendszerben, az előző törlése nélkül ábrázolja. Így több különböző ellenállású fogyasztó áramerősség - feszültség grafikonja tanulmányozható egyszerre.

A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával az addigi grafikonok törölhetők, és a program új ellenállást választ, a HELP lenyomásával pedig a program ismertetőjéhez lehet visszamenni.

A program a RUN/STOP billentyűvel állítható le. Vigyázni kell azonban arra, hogy a gép esetleg grafikus üzemmódban maradjon, így a READY felirat és a villogó kurzor nem jelenik meg. Ezt a GRAPHIC 0 parancs "vakon" történő beírásával és a RETURN gomb megnyomásával szüntethetjük meg. Elkerülhetjük ezt a helyzetet, ha az előzőekben leírtak szerint a HELP billentyű megnyomásával visszamegyünk a

bevezetőhöz, és itt állítjuk le a programot.  
A funkcióbillentyűk a program leállítás után is átprogramozott állapotban maradnak. Ez a helyzet a RESET gomb megnyomásával megszüntethető, de ez az eljárás a programot is törli.

```
10 rem ohm törvénye
20 rem zatonyi sandor bekescsaba 1985.
30 fori=1to7:keyi,"":next:key8,chr$(140)
40 graphic1:graphic0,1
50 color0,10,1;color1,10,7;color4,10,1
60 printchr$(14);chr$(8)
70 print"          U"
80 print"      R = ———      OHM TORVENYE"
90 print"          I"
100 print
110 print" A programmal a fogyasztora kapcsolt"
120 print" feszultseg es az aramosseg kapcso-"
130 print" lata vizsgalhato."
140 print:print" A feszultseg a felfele es lefele"
150 print" mutato nyilakkal változtathato."
160 print:print" A feszultseg es aramosseg kozti"
170 print" osszefuggest a program kirajzolja."
180 print:print" Az ellenallast az R betu megnyomasakor"
190 print" irja ki, az U lenyomasara uj"
200 print" fogyasztot ad."
210 print:print" a SHIFT es a CLEAR/HOME egyuttess"
220 print" megnyomasa torli az addig rajzoltakat."
230 char1,25,23,"TOVABB = ➤T"
240 getkeya$
250 ifa$<>"t"anda$<>"T"then240
260 color0,2,7;color1,1;color4,2,7
270 graphic1,1
280 box1,12,24,108,120
290 box1,32,92,88,120
300 draw0,46,120to74,120
310 box1,40,18,80,30
320 draw0,41,24to79,24
330 circle1,108,60,10
340 draw0,108,51to108,69
350 circle1,60,92,10
360 draw0,51,92to69,92
370 circle1,48,120,2
380 circle1,72,120,2
390 char1,7,15,"u"
400 char1,7,11,"v"
```

```

410 char1,7,4,"r"
420 char1,13,7,"a"
430 draw1,144,8to144,184to312,184
440 draw1,140,12to144,8to148,12
450 draw1,308,180to312,184to308,188
460 draw1,194,184to194,187
470 draw1,244,184to244,187
480 draw1,294,184to294,187
490 draw1,140,134to144,134
500 draw1,140,84to144,84
510 draw1,140,34to144,34
520 char1,19,2,"i":char1,38,21,"u"
530 char1,23,24,"50      100      150"
540 char1,15,5,"15":char1,15,11,"10":char1,16,17,"5"
550 r=5+int(50*rnd(0))
560 u=0
570 char1,5,18,"          "
580 char1,2,18,"u =" +str$(0.01*int(100*u))+" v"
590 i=u/r
600 char1,5,20,"          "
610 char1,2,20,"i =" +str$(0.01*int(100*i))+" a"
620 draw1,144+u,184-i*10
630 getkeya$
640 ifa$=" " then u=u+1
650 ifa$=" " then u=u-1
660 ifa$="r" then 720
670 ifa$=" " then 260
680 ifa$="u" then 760
690 ifa$=chr$(140) then run
700 if u<0 or u>150 or u/r>15 then 760
710 goto 570
720 char1,2,22,"r =" +str$(r):for c=15304 to 15311:read a:poke c,a:next
:restore
730 getkeya$
740 char1,6,22,"          "
750 goto 640
760 char1,(148+u)/8,(188-i*10)/8,str$(r)
770 goto 550
780 data 60,102,195,195,102,36,231,0

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 210 s (41 mp.)

Speciális karakter: @ → Ω

A program az  $R = U / I$  összefüggés feladatokban történő alkalmazását gyakoroltatja. A program elindítása után egy rövid ismertető jelenik meg. Ennek elolvasása után a T gomb lenyomásával léphetünk tovább. A megjelenő menü alapján választhatjuk ki, hogy melyik mennyiség kiszámítását akarjuk gyakorolni. A választáshoz csak a megfelelő számot kell lenyomni. Itt nyílik lehetőség arra is, hogy a 9 gomb lenyomásával az addig elért eredményeket kiírassuk a képernyőre. A HELP megnyomásával a program elején található bevezetőhöz lehet visszalépni.

A megfelelő mennyiség kiválasztása után egy feladat jelenik meg, melynek szövege, illetve adatai véletlenszerűen változnak. A HELP megnyomására a gép megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az elért eredménybe.

A választ a billentyűzet segítségével lehet beírni, és a RETURN billentyű megnyomásával kell lezárni. A válaszba a mértékegységet is be kell írni. A mértékegységet rövidítve (Ω, V, A) vagy teljes nevükkel (Ohm, Volt, Amper) is beírhatjuk. Az Ω a @ billentyűvel írható be.

A gép a választ elemzi, és hiba esetén annak jellegére utaló üzenet kiírása után megismétli a kérdést. Ha a második válasz is helytelen, a megoldást is kiírja a képernyőre. Helyes válasz esetén csak a HELYES felirat jelenik meg. Ezután az utolsó sorban megjelenő kiírásnak megfelelően az U billentyűvel új feladat kérhető, vagy az M gombbal a menühez térhetünk vissza, és más típusú feladatot választhatunk. A HELP billentyűvel ez esetben is a bevezetőhöz lehet visszamenni.

Az elért eredmények kiírását a menü alapján a 9 gomb lenyomásával kérhetjük. A képernyőn típusonként és összesítve is megjelenik a kapott feladatok száma, az elért pontszám és az eredmény (százalékban kifejezve). Egy feladatra 2 illetve 1 pont kapható aszerint, hogy az első vagy csak a második válasz volt a helyes.

A T billentyűvel a menühöz, a HELP segítségével pedig a bevezetőhöz lehet visszajutni.

A program a RUN/STOP billentyűvel állítható le. A karakterkészlet módosítása miatt a képernyő a leállítás után elkövetett esetleges hiba következtében olvashatatlanává válik. Ezen az F1 billentyű megnyomásával segíthetünk. Ezzel az eljárással a program is a memóriában marad. A gép a program leállítás után is a módosított karakterkészletet használja, és a funkcióbillentyűk is átprogramozott állapotban maradnak. Ez a helyzet a RESET gomb megnyomásával szüntethető meg, de ez a megoldás a programot is törli a memóriából.

```
10 rem ellenallas (gyak.1.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 graphic0,1:restore
70 for i=0to21:reada:poke3048+i,a:next
80 for i=0to18:reada:poke4072+i,a:next
90 sys3048
100 for i=0to7:reada:poke15360+i,a:next
110 poke65298,56:poke65299,56
120 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
130 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
140 data60,102,195,195,102,36,231,0
150 key1,"poke65298,56:poke65299,56"+chr$(13)
160 for i=2to7:keyi,"":next:key8,chr$(140)
170 col=10:gosubl060
180 printchr$(14);chr$(8)
190 print"
200 print"----- R=U/I ----- ELLENALLAS (gyak.1.)"
210 print"
220 print" A program az R = U / I osszefugges fel-"
230 print" adatokban torteno alkalmazasat gyakorol-tatja."
240 print" A valaszokba a mertekegysegeket is be kell irni."
250 print" A valaszt a RETURN gomb megnyomasaval kell lezarni."
260 print" A program jegyzi az elert eredmenyt."
270 for i=1to9
280 readk$(i),t(i),p1(i),p2(i),d1(i),d2(i)
290 next
300 char1,25,24,"TOVABB = T"
310 getkeya$:ifa$="t"ora$="T"then320:elseifa$=chr$(140)then60:els
e310
320 scncr
330 col=7:gosubl060
340 char1,5,3,"1. Feszultseg"
350 char1,20,4,""
360 char1,5,5,"2. Aramossege > kiszamitasa"
```

```

370 char1,20,6,"|"
380 char1,5,7,"3. Ellenallas  |"
390 char1,5,9,"4. Vegyes feladatok"
400 char1,5,13,"5. Feszultseg  |"
410 char1,20,14,"| kiszamitasa"
420 char1,5,15,"6. Aramosseg  >"
430 char1,20,16,"| szoveg alapjan"
440 char1,5,17,"7. Ellenallas  |"
450 char1,5,19,"8. Vegyes szoveges feladatok"
460 char1,5,23,"9. EREDMENYEK"
470 getkey$:ift$=chr$(140)then60
480 t=val(t$)
490 ift<1ort>9then470
500 ift=9then1730
510 ift=1theni=1+3*rnd(0)
520 ift=2theni=4+3*rnd(0)
530 ift=3theni=7+3*rnd(0)
540 ift=4thent=int(1+3*rnd(0)):goto510
550 ift=5theni=1+3*rnd(0):t=4
560 ift=6theni=4+3*rnd(0):t=5
570 ift=7theni=7+3*rnd(0):t=6
580 ift=8thent=int(5+3*rnd(0)):goto550
590 :
600 k$=k$(i):p1=p1(i):p2=p2(i):d1=d1(i):d2=d2(i)
610 scncclr:col=11:gosub1060
620 x=int(34+66*rnd(0))*10^(d1-2)
630 y=int(8*rnd(0)+2)*10^(d2-1)
640 z=x*y
650 x$=str$(x):ifx<1thenx$=" 0"+right$(x$,len(x$)-1)
660 y$=str$(y):ify<1theny$=" 0"+right$(y$,len(y$)-1)
670 z$=str$(z):ifz<1thenz$=" 0"+right$(z$,len(z$)-1)
680 r=0
690 ontgosub1100,1150,1200,1250,1310,1370
700 char1,7,7,"←"
710 a$="":gosub1440
720 ifa$=chr$(140)thenprint" ":r(t)=r(t)-r:goto920
730 h=0
740 ifabs(val(b$)-e)>.0001thenh=1
750 n=1000000
760 ifabs(val(b$)-e*n)<.0001andn<>1thenh=2
770 n=n/10:ifn>.0000001then760
780 lb=len(b$):l1=len(m1$):l2=len(m2$)
790 ifright$(b$,l1)<>m1$then810
800 iflb>l1+1thenifval(left$(b$,lb-l1-1))=val(b$)then830:else840
810 ifright$(b$,l2)<>m2$then830
820 iflb>l2+1thenifval(left$(b$,lb-l2-1))=val(b$)then830:else840
830 h=h+4
840 char1,0,10,""
850 if(hand1)=1thenprint" Szamitasi hiba"
860 if(hand2)=2thenprint" Nagysagrendi hiba"
870 if(hand4)=4thenprint" Mertekegység hiba"
880 ifh=0then1010
890 r=r+1:r(t)=r(t)+1

```



```

900 tt=2000:gosub1530
910 ifr=lthenscnclr:goto690
920 char1,0,10,chr$(27)+"q"
930 char1,0,12,chr$(27)+"q"
940 ontgosub1550,1610,1670,1550,1610,1670
950 char1,7,24,"UJ FELADAT =  U  MENU =  M"
960 getkeya$:ifa$=chr$(140)then60
970 ifa$="u"ora$="U"then480
980 ifa$="m"ora$="M"then320
990 goto960
1000 :
1010 j(t)=j(t)+2-r
1020 char1,7,14,"Helyes."
1030 tt=1000:gosub1530
1040 goto950
1050 :
1060 :
1070 color0,col,1:color1,col,7:color4,col,1
1080 return
1090 :
1100 char1,1,1,"R ="+"x$+" @"
1110 char1,1,3,"I ="+"y$+" A"
1120 char1,1,5,"_____ "
1130 char1,1,7,"U = ? ":gotol290
1140 return
1150 char1,1,1,"U ="+"z$+" V"
1160 char1,1,3,"R ="+"x$+" @"
1170 char1,1,5,"_____ "
1180 char1,1,7,"I = ? ":gotol350
1190 return
1200 char1,1,1,"U ="+"z$+" V"
1210 char1,1,3,"I ="+"y$+" A"
1220 char1,1,5,"_____ "
1230 char1,1,7,"R = ? ,":gotol410
1240 return
1250 char1,0,1,k$
1260 char1,0,5,"Mekkora a ra kapcsolt feszultseg?"
1270 char1,p1,1,x$
1280 char1,p2,1,y$
1290 e=z:m1$="V":m2$="volt"
1300 return
1310 char1,0,1,k$
1320 char1,0,5,"Mekkora aram folyt at rajta?"
1330 char1,p1,1,z$
1340 char1,p2,1,x$
1350 e=y:m1$="A":m2$="amper"
1360 return
1370 char1,0,1,k$
1380 char1,0,5,"Mekkora az ellenallasa?"
1390 char1,p1,1,z$
1400 char1,p2,1,y$
1410 e=x:m1$="@" :m2$="ohm"
1420 return

```

```

1430 :
1440 b$=""
1450 getkeya$:ifa$=chr$(140)thenreturn
1460 ifa$=chr$(13)andb$<>" "thenprint" ":return
1470 ifa$=chr$(20)andb$>" "thenprinta$;"← ███";:b$=left$(b$,len(b$)
-1):gotol450
1480 if(a$<="9"anda$>="0")or(a$<="z"anda$>="@")or(a$<="Z"anda$>="
✓")then1510
1490 ifa$=". "ora$="*"ora$="/"then1510
1500 gotol450
1510 printa$;"← ███";:b$=b$+a$:iflen(b$)>15thenprint" ":return
1520 gotol450
1530 :
1540 fortt=ttto0step-1:next:return
1550 :
1560 char1,7,12,"U = R * I"
1570 char1,7,14,"U =" +x$+" @ *"+y$+" A"
1580 char1,7,16,"U =" +z$+" @*A"
1590 char1,7,18,"U =" +z$+" V"
1600 return
1610 :
1620 char1,7,12,"I = U / R"
1630 char1,7,14,"I =" +z$+" V /"+x$+" @"
1640 char1,7,16,"I =" +y$+" V/@"
1650 char1,7,18,"I =" +y$+" A"
1660 return
1670 :
1680 char1,7,12,"R = U / I"
1690 char1,7,14,"R =" +z$+" V /"+y$+" A"
1700 char1,7,16,"R =" +x$+" V/A"
1710 char1,7,18,"R =" +x$+" @"
1720 return
1730 :
1740 scnclr
1750 col=2:gosub1060
1760 char1,17,0,"feladat pont eredmeny"
1770 char1,0,3,"Feszultseg"
1780 char1,0,5,"Aramerosseg"
1790 char1,0,7,"Ellenallas"
1800 char1,0,11,"Feszultseg (szov.)"
1810 char1,0,13,"Aramerosseg (szov.)"
1820 char1,0,15,"Ellenallas (szov.)"
1830 char1,0,19,"OSSZESEN"
1840 fori=1to3
1850 j=j(i):n=j+r(i)
1860 char1,19,1+2*i,"":printusing"###";n/2
1870 char1,26,1+2*i,"":printusing"###";j
1880 ifn=0then1910
1890 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1900 char1,34,1+2*i,"":printusing"### %";j*100/n
1910 color1,2:next
1920 fori=4to6
1930 j=j(i):n=j+r(i)

```

```

1940 char1,19,3+2*i,"":printusing"###";n/2
1950 char1,26,3+2*i,"":printusing"###";j
1960 ifn=0then1990
1970 ifj/n>.8thencolor1,3,4:elseifj/n<.5thencolor1,7,6
1980 char1,34,3+2*i,"":printusing"### %";j*100/n
1990 color1,2:next
2000 color1,8,6
2010 j=0:r=0:fori=1to6:j=j+j(i):r=r+r(i):next:n=r+j
2020 char1,19,19,"":printusing"###";n/2
2030 char1,26,19,"":printusing"###";j
2040 ifn=0then2060
2050 char1,34,19,"":printusing"### %";j*100/n
2060 color1,2
2070 goto300
2080 data"Egy @-os ellenallason A erossegu aram halad at.
"
2090 data1,3,24,2,0
2100 data"Egy @ ellenallasu fogyaszton A erossegu aram
folyik."
2110 data1,3,32,3,1
2120 data"Egy @ ellenallasu izzon A-es aram folyik ke
resztul."
2130 data1,3,26,2,0
2140 data"Egy @-os fogyasztot V feszultsegre k
apcsoltak."
2150 data2,23,3,3,0
2160 data"Egy @-os ellenallasra V feszultseget k
apcsoltunk."
2170 data2,27,3,5,-1
2180 data"Egy @-os izzon V feszultseget mertunk."
2190 data2,17,3,2,0
2200 data"Egy V-ra kapcsolt lampan A erossegu aram
folyik."
2210 data3,3,29,2,0
2220 data"Egy ellenallason V, illetve A merheto."
2230 data3,16,31,3,0
2240 data"Egy fogyaszton V hatasara A erossegu aram
folyik."
2250 data3,14,30,3,1

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás			
ellenőrzés			

Betöltési idő: 210 s (41 mp.)

Speciális karakter: @ → Ω

A program az ellenállás fogalmának elmélyítéséhez készült. Az  $R = U / I$  összefüggésben szereplő három mennyiség közül a gép 2 - 2 mennyiség nagyságrendi viszonyait megadja, és ennek ismeretében kell a tanulónak a hiányzó mennyiségeket összehasonlítani. (Pl. a gép megadja, hogy a feszültségek egyenlők, az egyik fogyasztón pedig kétszer nagyobb áram folyik keresztül. Ebben az esetben a két ellenállást kell összehasonlítani.)

Elindítás után egy rövid tájékoztatót ad a program. Ezt elolvasva a T lenyomásával lehet továbblépni. Ezután történik a nehézségi fokozat kiválasztása. A felsorolt lehetőségek közül a megfelelő szám lenyomásával lehet választani. Itt van mód arra, hogy az addig elért eredményt a 4 gomb lenyomásával kiírassuk a képernyőre. A HELP gomb lenyomásával a bevezetőhöz lehet visszamenni.

A nehézségi fokozat meghatározása után az összehasonlítandó mennyiséget kell kiválasztani. Itt is a megfelelő szám lenyomásával lehet választani. Ezután a program a választásokat figyelembe véve véletlenszerű értéket ad 2 - 2 mennyiségnek, és rákérdez a hiányzó mennyiségek összehasonlításának eredményére. A választ a < = > és a ? gombok segítségével lehet beírni.. A ? gombot akkor kell lenyomni, ha a kérdésre nem adható egyértelmű válasz. A HELP gomb megnyomására a program megadja a helyes megoldást, de ekkor ezt a feladatot nem számítja be az eredménybe. A válasz beírása után, ha az hibás volt, a program az erre utaló üzenet kiírása után megadja a helyes választ, ellenkező esetben csak a REMEK feirat jelenik meg a képernyőn. Ezután az U gomb megnyomásával új feladatot kérhetünk, az M megnyomásával pedig az első menühez (nehézségi fokozat kiválasztása és eredmények) mehetünk vissza. A HELP megnyomása ebben az esetben a bevezetőhöz való visszatérést eredményezi.

Az eredmények kiírását az említett módon (az első menü alapján, a 4 gomb lenyomásával) kérhetjük. A program nehézségi fokozat és az

összehasonlítandó mennyiség szerint csoportosítva, valamint összesítve százalékban adja meg az elért eredményt. Az eredmények kiírása után a T gombbal újra az első menühöz lehet visszajutni, míg a HELP lenyomásával a bevezető ismertetéshez léphetünk vissza. A program a RUN/STOP billentyűvel állítható le. A karakterkészlet átírása miatt a leállítás után elkövetett esetleges hiba következtében a képernyő olvashatatlaná válhat. Ezen az F1 gomb lenyomásával segíthetünk, és ezzel a program sem törlődik. A gép a program leállítása után is a módosított karakterkészlettel dolgozik és a funkcióbillentyűk is átállított állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez a program törlését eredményezi.

```

10 rem ellenallas (gyak 2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnclr
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next
110 poke65298,56:poke65299,56
120 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
130 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
140 data60,102,195,195,102,36,231,0
150 key1,"poke65298,56:poke65299,56"+chr$(13)
160 fori=2to7:keyi,"":next:key8,chr$(140)
170 fok$(1)="Konnyu ":t$(1)="Feszultseg"
180 fok$(2)="Kozepes":t$(2)="Aramerosseg"
190 fok$(3)="Nehez ":t$(3)="Ellenallas"
200 t$(4)="VEGYES"
210 col=11:gosub2140
220 printchr$(14);chr$(8)
230 print"      □      □"
240 print"——□ R=U/I □—— ELLENALLAS (gyak.2)"
250 print"      □      □"
260 print"□□□□A program az R = U / I osszefugges"
270 print"elemzesehez keszult."
280 print"□A megadott adatok alapjan ossze kell"
290 print"hasonlitani a ket mennyiseget."
300 print"□Az osszehasonlitas eredmenyet a < = >"
310 print"ombokkal lehet beirni."
320 print"□Ha nem adható egyertelmu valasz, akkor"
330 print"a ? gombot kell megnyomni."
340 char1,25,24,"TOVABB = □□"
350 getkeya$:ifa$=chr$(140)thenrun
360 ifa$<>"t"anda$<>"T"then350

```

```

370 scnclr
380 col=10:gosub2140
390 char1,5,0,"NEHEZSEGI FOKOZAT VALASZTAS"
400 char1,14,7,"1. "+fok$(1)
410 char1,14,10,"2. "+fok$(2)
420 char1,14,13,"3. "+fok$(3)
430 char1,14,18,"4. EREDMENYEK"
440 char1,2,24,"A szam lenyomasaval lehet valasztani."
450 getkeya$:ifa$=chr$(140)thenrun
460 ifa$="4"then1860
470 ifa$<"1"ora$>"3"then450
480 f=val(a$)
490 scnclr
500 col=10:gosub2140
510 gosub2170
520 char1,4,7,"AZ OSSZEHASONLITANDO MENNYISEG"
530 char1,13,12,"1. "+t$(1)
540 char1,13,14,"2. "+t$(2)
550 char1,13,16,"3. "+t$(3)
560 char1,13,20,"4. "+t$(4)
570 char1,2,24,"A szam lenyomasaval lehet valasztani."
580 getkeyt$:ift$=chr$(140)thenrun
590 ift$<"1"ort$>"4"then580
600 t=val(t$):ift=4thent=1+3*rnd(0)
610 scnclr
620 col=11:gosub2140
630 gosub2170
640 x1=int(1+9*rnd(0)):x2=int(1+9*rnd(0))
650 y1=int(1+9*rnd(0)):y2=int(1+9*rnd(0))
660 r=rnd(0)
670 ifr<.3then700
680 ifr<.65thenx1=x2
690 ifr>=.65theny1=y2
700 ontgosub930,1240,1550
710 char1,8,5,x1$:char1,27,5,x2$
720 char1,8,9,y1$:char1,27,9,y2$
730 char1,7,11,"-----"
740 char1,8,13,z1$:char1,27,13,z2$
750 char1,19,13,"??"
760 getkeya$:ifa$=chr$(140)then860
770 ifa$=","ora$="."ora$="/"thena$=chr$(asc(a$)+16)
780 ifa$<>"<"anda$<>"="anda$<>">"anda$<>"?"then760:elsechar1,19,1
3,a$
790 n(f,t)=n(f,t)+1
800 ifa$=e$then810:else840
810 j(f,t)=j(f,t)+1
820 char1,17,18,"REMEK"
830 goto880
840 char1,17,16,"ROSSZ"
850 fort=0to300:next
860 char1,11,20,"A helyes valasz: ?"+e$+" "
870 fort=0to900:next
880 char1,5,24,"UJ FELADAT = U          MENU = M"

```

```

890 getkeya$:ifa$=chr$(140)thenrun
900 ifa$="m"ora$="M"then370
910 ifa$="u"ora$="U"then600
920 goto890
930 :
940 z1$=" U1":z2$=" U2"
950 onfgoto960,1010,1120
960 :
970 x1$=str$(x1)+"@":x2$=str$(x2)+"@"
980 y1$=str$(y1)+"A":y2$=str$(y2)+"A"
990 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
"
1000 return
1010 :
1020 x1$=str$(x1)+"R":x2$=str$(x2)+"R"
1030 ifx1=1thenx1$=" R"
1040 ifx2=1thenx2$=" R"
1050 ifx1=x2thenx1$=" R":x2$=x1$
1060 y1$=str$(y1)+"I":y2$=str$(y2)+"I"
1070 ify1=1theny1$=" I"
1080 ify2=1theny2$=" I"
1090 ify1=y2theny1$=" I":y2$=y1$
1100 ifx1*y1<x2*y2thene$="<":elseifx1*y1=x2*y2thene$="=":elsee$=">"
">"
1110 return
1120 :
1130 x1$=" R1":x2$=" R2"
1140 y1$=" I1":y2$=" I2"
1150 ifx1=x2andyl=y2thene$="=":gotol190
1160 ifx1<=x2andyl<=y2thene$="<":gotol190
1170 ifx1>=x2andyl>=y2thene$=">":gotol190
1180 e$="?"
1190 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1200 char1,19,5,r$
1210 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1220 char1,19,9,r$
1230 return
1240 :
1250 z1$=" I1":z2$=" I2"
1260 onfgotol280,1320,1430
1270 :
1280 x1$=str$(x1)+"V":x2$=str$(x2)+"V"
1290 y1$=str$(y1)+"@":y2$=str$(y2)+"@"
1300 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$=">"
">"
1310 return
1320 :
1330 x1$=str$(x1)+"U":x2$=str$(x2)+"U"
1340 ifx1=1thenx1$=" U"
1350 ifx2=1thenx2$=" U"
1360 ifx1=x2thenx1$=" U":x2$=x1$
1370 y1$=str$(y1)+"R":y2$=str$(y2)+"R"
1380 ify1=1theny1$=" R"

```

```

1390 ify2=1theny2$=" R"
1400 ify1=y2theny1$=" R":y2$=y1$
1410 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1420 return
1430 :
1440 x1$=" U1":x2$=" U2"
1450 y1$=" R1":y2$=" R2"
1460 ifx1=x2andy1=y2thene$="=":goto1500
1470 ifx1<=x2andy1>=y2thene$="<":goto1500
1480 ifx1>=x2andy1<=y2thene$=">":goto1500
1490 e$="?"
1500 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1510 char1,19,5,r$
1520 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1530 char1,19,9,r$
1540 return
1550 :
1560 z1$=" R1":z2$=" R2"
1570 onfgoto1580,1630,1740
1580 :
1590 x1$=str$(x1)+"V":x2$=str$(x2)+"V"
1600 y1$=str$(y1)+"A":y2$=str$(y2)+"A"
1610 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1620 return
1630 :
1640 x1$=str$(x1)+"U":x2$=str$(x2)+"U"
1650 ifx1=1thenx1$=" U"
1660 ifx2=1thenx2$=" U"
1670 ifx1=x2thenx1$=" U":x2$=x1$
1680 y1$=str$(y1)+"I":y2$=str$(y2)+"I"
1690 ify1=1theny1$=" I"
1700 ify2=1theny2$=" I"
1710 ify1=y2theny1$=" I":y2$=y1$
1720 ifx1/y1<x2/y2thene$="<":elseifx1/y1=x2/y2thene$="=":elsee$="
>"
1730 return
1740 :
1750 x1$=" U1":x2$=" U2"
1760 y1$=" I1":y2$=" I2"
1770 ifx1=x2andy1=y2thene$="=":goto1810
1780 ifx1<=x2andy1>=y2thene$="<":goto1810
1790 ifx1>=x2andy1<=y2thene$=">":goto1810
1800 e$="?"
1810 ifx1<x2thenr$="<":elseifx1=x2thenr$="=":elser$=">"
1820 char1,19,5,r$
1830 ify1<y2thenr$="<":elseify1=y2thenr$="=":elser$=">"
1840 char1,19,9,r$
1850 return
1860 :
1870 col=15:gosub2140
1880 scnclr
1890 char1,15,0,"EREDMENYEK"

```



```

1900 char1,14,5,"1.      2.      3.      OSSZ."
1910 char1,0,9,t$(1)
1920 char1,0,12,t$(2)
1930 char1,0,15,t$(3)
1940 char1,0,19,"OSSZESEN"
1950 fort=1to3
1960 n(4,t)=0:j(4,t)=0
1970 forf=1to3
1980 ifn(f,t)>0thene(f,t)=100*j(f,t)/n(f,t)
1990 n(4,t)=n(4,t)+n(f,t)
2000 j(4,t)=j(4,t)+j(f,t)
2010 next
2020 ifn(4,t)>0thene(4,t)=100*j(4,t)/n(4,t)
2030 next
2040 forf=1to4
2050 n(f,4)=0:j(f,4)=0
2060 fort=1to3
2070 ifn(f,t)>0thenchar1,5+7*f,6+3*t,"":printusing"###%";e(f,t)
2080 n(f,4)=n(f,4)+n(f,t)
2090 j(f,4)=j(f,4)+j(f,t)
2100 next
2110 ifn(f,4)>0thenchar1,5+7*f,19,"":printusing"###%";100*j(f,4)/
n(f,4)
2120 next
2130 goto340
2140 :
2150 color0,col,1:color1,col,7:color4,col,1
2160 return
2170 :
2180 char1,0,0," "
2190 char1,0,1," "      Fokozat: "+fok$(f)+"      "
2200 char1,0,2," "      "
2210 return

```

	osztály	csoport	egyéni
új ismeret	○	○	●
alkalmazás			●
ellenőrzés			

Betöltési idő: 200 s (40 mp.)

Speciális karakter: @ → Ω

A program az  $R = U / I$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább egy menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, amelynek értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésköznek 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységeket nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól. Ezekután három lehetőség közül választhatunk. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez térhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN megnyomásával az előzőleg használt értéket megtartsuk. A program futása a RUN/STOP billentyű megnyomásával szakítható meg. Mivel a gép a program leállítása után is a módosított karakterkészletet használja, egy esetleges hiba a képernyő olvashatatlaná válását eredményezi. Ez az állapot az f1 gomb megnyomásával megszüntethető. A funkcióbillentyűk átprogramozott állapota és a karakterkészlet a RESET gomb megnyomásával állítható vissza az eredeti állapotba, de ez a program törlését eredményezi.

```

10 rem ellenallas (tabl.2.)
20 rem zatonyi sandor bekescsaba 1986.
30 poke55,0:poke56,56
40 poke53,0:poke54,56
50 poke51,0:poke52,56
60 scnc1r
70 fori=0to21:reada:poke3048+i,a:next
80 fori=0to18:reada:poke4072+i,a:next
90 sys3048
100 fori=0to7:reada:poke15360+i,a:next:rem @
110 poke65298,56:poke65299,56
120 data 160,8,162,255,189,0,208,157,0,56,202,208,247,173,0,208,1
41,0,56,76,232
130 data 15,238,238,11,238,241,11,238,247,11,238,250,11,136,208,1
,96,76,234,11
140 data60,102,195,195,102,36,231,0
150 key1,"poke65298,56:poke65299,56"+chr$(13)
160 fori=2to7:keyi,"":next:key8,chr$(140)
170 f1$=" FESZULTSEG ARAMEROSSEG ELLENALLAS"
180 f2$=" (V) (A) (0)"
190 v1$="FESZULTSEG"
200 v2$="ARAMEROSSEG"
210 v3$="ELLENALLAS"
220 v$="-----"
230 sp$=" "
240 col=11:gosub1990
250 printchr$(14);chr$(8)
260 print" "
270 print" R=U/I ELLENALLAS (tabl.2)"
280 print" "
290 print" A program az R = U / I osszefugges elem-"
300 print" zesehez keszult."
310 print" A három mennyiség közül egyet allando"
320 print" értéknek választva egy másik mennyiség"
330 print" értéke szabadon változtatható."
340 print" A program ezek figyelembevételével"
350 print" kiszámítja a harmadik változó megfelelő"
360 print" értékeit."
370 print" A CLEAR/HOME megnyomására új értékek"
380 print" adhatók az egyes mennyiségeknek, a"

```

```

390 print"SHIFT es CLEAR/HOME egyutt es lenyomasa"
400 print"utan pedig az egyes mennyisegek szerepe"
410 print"valtoztathato meg."
420 char1,25,24,"TOVABB = T"
430 getkeya$:ifa$=chr$(140)thenrun
440 ifa$<"t"anda$<"T"then430
450 col=10:gosub1990
460 all=50:kezd=10:lep=1
470 print" " +sp$+ "          A VIZSGALT ESET KIVALASZTASA          "+sp$
480 char1,7,5,"FUGGO          FUGGETLEN          "
490 char1,18,6,"ALLANDO"
500 char1,6,7,"VALTOZO          VALTOZO          "
510 char1,0,8,v$
520 char1,0,9," 1. FESZULTSEG  ARAMEROSSEG  ELLENALLAS"+sp$
530 char1,0,11," 2. FESZULTSEG  ELLENALLAS  ARAMEROSSEG"+v$
540 char1,0,13," 3. ARAMEROSSEG  FESZULTSEG  ELLENALLAS "+sp$
550 char1,0,15," 4. ARAMEROSSEG  ELLENALLAS  FESZULTSEG "+v$
560 char1,0,17," 5. ELLENALLAS  FESZULTSEG  ARAMEROSSEG"+sp$
570 char1,0,19," 6. ELLENALLAS  ARAMEROSSEG  FESZULTSEG "+v$
580 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET          VALASZTA
NI."
590 getkeyt$:ift$=chr$(140)thenrun
600 ift$<"1"ort$>"6"then590
610 col=15:gosub1990
620 print" " ;sp$;"          ERTEKADAS          " ;sp$
630 char1,2,5,"FUGGO VALTOZO          :"
640 char1,2,7,"Allando mennyiseg          :"
650 char1,2,9,"Fuggetlen valtozo          :"
660 onval(t$)goto670,850,1030,1210,1390,1570
670 :
680 char1,22,5,v1$
690 char1,22,7,v2$
700 char1,22,9,v3$
710 char1,0,11,v$
720 char1,0,14,"Az aramosseg erteke (A) : "
730 a=all:p=28:q=14:gosub1750:all=a
740 char1,0,17,"Az ellenallas"
750 char1,5,19,"- kezdoerteke (@) : "
760 a=kezd:p=25:q=19:gosub1750:kezd=a
770 char1,5,21,"- lepeskoz (@) : "
780 a=lep:p=22:q=21:gosub1750:lep=a
790 col=9:gosub1990
800 char1,2,1," T"
810 forj=0to9
820 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
830 next
840 goto1880
850 :
860 char1,22,5,v1$
870 char1,22,7,v3$
880 char1,22,9,v2$
890 char1,0,11,v$
900 char1,0,14,"Az ellenallas erteke (@) : "

```

```

910 a=all:p=27:q=14:gosub1750:all=a
920 char1,0,17,"Az aramosseg"
930 char1,5,19,"- kezdoerteke (A) : "
940 a=kezd:p=25:q=19:gosub1750:kezd=a
950 char1,5,21,"- lepeskoz (A) : "
960 a=lep:p=22:q=21:gosub1750:lep=a
970 col=9:gosub1990
980 char1,2,1," ┌───────────┐"
990 forj=0to9
1000 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
1010 next
1020 goto1880
1030 :
1040 char1,22,5,v2$
1050 char1,22,7,v1$
1060 char1,22,9,v3$
1070 char1,0,11,v$
1080 char1,0,14,"A feszultseg erteke (V) : "
1090 a=all:p=26:q=14:gosub1750:all=a
1100 char1,0,17,"Az ellenallas"
1110 char1,5,19,"- kezdoerteke (Q) : "
1120 a=kezd:p=25:q=19:gosub1750:kezd=a
1130 char1,5,21,"- lepeskoz (Q) : "
1140 a=lep:p=22:q=21:gosub1750:lep=a
1150 col=9:gosub1990
1160 char1,14,1," ┌───────────┐"
1170 forj=0to9
1180 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1190 next
1200 goto1880
1210 :
1220 char1,22,5,v2$
1230 char1,22,7,v3$
1240 char1,22,9,v1$
1250 char1,0,11,v$
1260 char1,0,14,"A feszultseg erteke (V) : "
1270 a=all:p=26:q=14:gosub1750:all=a
1280 char1,0,17,"Az ellenallas"
1290 char1,5,19,"- kezdoerteke (Q) : "
1300 a=kezd:p=25:q=19:gosub1750:kezd=a
1310 char1,5,21,"- lepeskoz (Q) : "
1320 a=lep:p=22:q=21:gosub1750:lep=a
1330 col=9:gosub1990
1340 char1,14,1," ┌───────────┐"
1350 forj=0to9
1360 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1370 next
1380 goto1880
1390 :
1400 char1,22,5,v3$
1410 char1,22,7,v1$
1420 char1,22,9,v2$
1430 char1,0,11,v$

```

```

1440 char1,0,14,"A feszultseg erteke (V) : "
1450 a=all:p=26:q=14:gosub1750:all=a
1460 char1,0,17,"Az aramosseg"
1470 char1,5,19,"- kezdoerteke (A) : "
1480 a=kezd:p=25:q=19:gosub1750:kezd=a
1490 char1,5,21,"- lepeskoz (A) : "
1500 a=lep:p=22:q=21:gosub1750:lep=a
1510 col=9:gosub1990
1520 char1,28,1," ────────────┘"
1530 forj=0to9
1540 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1550 next
1560 goto1880
1570 :
1580 char1,22,5,v3$
1590 char1,22,7,v2$
1600 char1,22,9,v1$
1610 char1,0,11,v$
1620 char1,0,14,"Az aramosseg erteke (A) : "
1630 a=all:p=28:q=14:gosub1750:all=a
1640 char1,0,17,"A feszultseg"
1650 char1,5,19,"- kezdoerteke (V) : "
1660 a=kezd:p=25:q=19:gosub1750:kezd=a
1670 char1,5,21,"- lepeskoz (V) : "
1680 a=lep:p=22:q=21:gosub1750:lep=a
1690 col=9:gosub1990
1700 char1,28,1," ────────────┘"
1710 forj=0to9
1720 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1730 next
1740 goto1880
1750 :
1760 b$="":char1,p,q," "
1770 print" ←┘";:getkeya$
1780 ifa$=chr$(13)then1820
1790 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$a$
1800 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1810 iflen(b$)<6then1770
1820 ifb$=""then1850
1830 ifval(b$)<.01thenchar1,p,q,sp$:goto1750
1840 a=val(b$)
1850 char1,p,q,str$(a)+sp$
1860 ifa<1thenchar1,p,q,"0"
1870 return
1880 :
1890 char1,0,0,f1$:char1,0,3,f2$
1900 forj=0to9

```

```
1910 char1,0,6+2*j," "  
1920 printusing"#####.#";x(j);  
1930 printusing"#####.#";y(j);  
1940 printusing"#####";z(j);  
1950 next  
1960 getkeya$:ifa$="☹"then610:elseifa$=""then450:elseifa$=chr$(14  
0)thenrun  
1970 goto1960  
1980 stop  
1990 :  
2000 scnclr  
2010 color0,col,6:color1,col,1:color4,col,6  
2020 return
```

	osztály	csoport	egyéni
új ismeret			
alkalmazás	○	○	●
ellenőrzés			●

Betöltési idő: 200 s (40 mp.)

Speciális karakter: [ → | ] → | ● → 7

€ → 7 ☑ → + ☒ → + ☒ → Ω

A program két fogyasztó kapcsolásának felismerését és az eredő ellenállás meghatározását gyakoroltatja. Indítás után egy rövid ismertetőt ad a program. Ennek elolvasása után a T gomb lenyomásával mehetünk tovább.

A képernyőn véletlen választás alapján két sorba - vagy párhuzamosan kötött fogyasztó áramköri rajza jelenik meg. A rajz alatt a gép feltünteti a fogyasztók ellenállását. Az eredő ellenállás meghatározása után a felsorolt válaszlehetőségek közül kell a helyeset kiválasztani, és a neki megfelelő betűt megnyomni a billentyűzeten. Ha a válasz helyes volt, a REMEK felirat jelenik meg, egyébként a ROSSZ üzenettel jelzi a gép a hibás választ. Helytelen válasz esetén a jó megoldás előtt álló betű villog, jelezve, hogy mit kellett volna válaszolni. A HELP megnyomására a gép ugyanúgy jelzi a jó választ, de ekkor ezt a feladatot nem számítja be az eredménybe.

A helyes, illetve a rossz válaszok számát a gép a képernyő bal alsó sarkában összesítve írja ki. A jobb alsó sarokban az eltelt gondolkodási időt jelzi a gép óra: perc: másodperc: alakban. A következő feladatot a T gomb megnyomásával lehet kérni. A HELP megnyomására ilyenkor a program a bevezető menühez megy vissza, de az addigi eredményeket törli. A SHIFT és a CLEAR/HOME megnyomásával szintén új feladat kérhető, de ilyenkor az elért eredmény és a gondolkodási idő is törlődik.

A program a RUN/STOP billentyű megnyomásával állítható meg. A gép a program leállítása után is a megváltozott karakterkészletet használja. Ennek következtében egy esetleges hiba miatt a képernyő olvashatatlaná válik. Az f1 billentyű lenyomásával a képernyő újra olvasható lesz. Az eredeti karakterkészlet a RESET gomb megnyomásával állítható vissza, és ez a funkcióbillentyűket is az eredeti állapotba hozza, azonban a programot kitörli a memóriából.





```

490 j=1
500 gosub950
510 print"R <;r1;"**"
520 gosub950
530 print"R =";r1;"**"
540 gosub950
550 ifr1=r2then600
560 print"R < R <;r2;"**";;ifh$="d"thenh$="f"
570 gosub950
580 print"R =";r2;"**"
590 gosub950
600 print"R < R <;r1+r2;"**";
610 gosub950
620 print"R ,=";r1+r2;"**"
630 gosub950
640 print"R >;r1+r2;"**"
650 char1,2,24,"A betu lenyomasaval lehet valaszolni."
660 ti$=t$
670 geta$:a=asc(a$)and127
680 char1,31,22,left$(ti$,2)+":"+mid$(ti$,3,2)+":"+right$(ti$,2)
690 if(a<65ora>64+j)anda<>8then670
700 t$=ti$
710 char1,1,24,chr$(27)+"q"
720 ifa=8then850
730 ifa<>asc(h$)then800
740 :
750 char1,16,22,"K-----@"
760 char1,16,23," | REMEK |"
770 char1,16,24," |-----|"
780 h=h+1
790 goto880
800 :
810 char1,16,22,"K-----@"
820 char1,16,23," | ROSSZ |"
830 char1,16,24," |-----|"
840 ro=ro+1
850 :
860 j=asc(h$)-64
870 print"R";:gosub950
880 char1,0,22,"Helyes :"+str$(h)
890 char1,0,24,"Rossz :"+str$(ro)
900 char1,29,24,"TOVABB = T"
910 geta$:ifa$=chr$(8)then240
920 ifa$="R"thenh=0:ro=0:goto410
930 ifa$="t"ora$="T"then410
940 goto910
950 :
960 char1,19,3*j-3,"K-----@"
970 char1,19,3*j-2," | |"
980 char1,19,3*j-1," |-----|"
990 char1,20,3*j-2,chr$(96+j)
1000 char1,24,3*j-2,"R"
1010 j=j+1

```

```

1020 return
1030 : rem (1)
1040 print" £——① "
1050 print" £——[ ]——①"
1060 print" |      |      |"
1070 print" |      |      |"
1080 print" |      |      |"
1090 print" |      |      |"
1100 print" |      £——① |"
1110 print" ]——[ ]——[ "
1120 print" |      |      |"
1130 print" |      |      |"
1140 print" |      |      |"
1150 print" |      |      |"
1160 print" |      ① ① |"
1170 return
1180 : rem (2)
1190 print" £——① "
1200 print" £——[ ]——① "
1210 print" |      |      |"
1220 print" |      |      |"
1230 print" |      |      £√①"
1240 print" |      |      |"
1250 print" |      |      |"
1260 print" |      |      |"
1270 print" |      |      |"
1280 print" |      |      √"
1290 print" |      |      |"
1300 print" |      |      |"
1310 print" |      ① ① |"
1320 return
1330 : rem (3)
1340 print" £——√——① "
1350 print" |      |      |"
1360 print" |      |      |"
1370 print" |      |      |"
1380 print" |      £√①  £√①"
1390 print" |      |      |"
1400 print" |      |      |"
1410 print" |      |      |"
1420 print" |      |      |"
1430 print" |      √      √"
1440 print" |      |      |"
1450 print" |      |      |"
1460 print" |      |      |"
1470 return
1480 : rem (4)
1490 print" £——① "
1500 print" |      |"
1510 print" |      |"
1520 print" |      |"
1530 print" £√①  £√①"
1540 print" |      |"

```

```

1550 print" | | | |"
1560 print" | | | |"
1570 print" | | | |"
1580 print" // | // "
1590 print" | | | |"
1600 print" | | | |"
1610 print" | | | |"
1620 return
1630 : rem (5)
1640 print" £——//——@ "
1650 print" | | | |"
1660 print" | | | |"
1670 print" | | | |"
1680 print" £v@ | £v@"
1690 print" | | | |"
1700 print" | | | |"
1710 print" | | | |"
1720 print" | | | |"
1730 print" // | // "
1740 print" | | | |"
1750 print" | | | |"
1760 print" | | | |"
1770 return
1780 : rem (6)
1790 print" £——@ "
1800 print" £——[ ]——@"
1810 print" | | | |"
1820 print" | | | |"
1830 print" | | | |"
1840 print" | | | |"
1850 print" | | | |"
1860 print" | | | |"
1870 print" | | | |"
1880 print" | | | |"
1890 print" | | | |"
1900 print" | | £——@ |"
1910 print" | | [ ] |"
1920 print" | | | |"
1930 return
1940 : rem (7)
1950 print" £——o o——@"
1960 print" | | | |"
1970 print" | | | |"
1980 print" | | | |"
1990 print" | | £——@ |"
2000 print" ]——[ ]——["
2010 print" | | | |"
2020 print" | | | |"
2030 print" | | | |"
2040 print" | | | |"
2050 print" | | £——@ |"
2060 print" | | [ ] |"
2070 print" | | | |"

```

```

2080 return
2090 : rem (6)
2100 print"      £———Ⓞ      "
2110 print" £——[      ]——Ⓞ"
2120 print" |      [      ]      |"
2130 print" |      |      |"
2140 print" |      |      |"
2150 print" |      |      |"
2160 print" |      |      |"
2170 print" |      |      |"
2180 print" |      |      |"
2190 print" |      |      |"
2200 print" |      |      |"
2210 print" |      £———Ⓞ      |"
2220 print" |——[      ]——|"
2230 print" |      [      ]      |"
2240 return

```

	osztály	csoport	egyéni
új ismeret			
alkalmazás			
ellenőrzés			

Betöltési idő: 195 s (39 mp.)

Speciális karakter: nincs

A program a  $P = U \cdot I$  összefüggés vizsgálatához készült. A képletben szereplő három mennyiség közül egyet állandó értékűnek választva a két másik mennyiség közti viszony táblázatos formában tanulmányozható. Ehhez az egyik mennyiséget tetszőleges módon változtatva a program kiszámítja a másik mennyiség megfelelő értékeit.

Az indítás után egy ismertető jelenik meg a képernyőn. Ezt elolvasva a T gomb lenyomásával léphetünk tovább a menühez. Itt dönthetjük el, hogy a három mennyiség közül melyik értéke legyen állandó, melyiket akarjuk független változóként használni és melyik legyen a függő változó, amelynek értékeit a program számítja ki az előző két változó értékeiből. A választáshoz csupán a megfelelő gombot kell lenyomni. A HELP billentyű megnyomásával a bevezetőhöz mehetünk vissza.

A program a megfelelő választás után rákérdez az állandónak választott mennyiség értékére. Ezután a független változó értékeit kell megadni úgy, hogy megadjuk a kezdőértéket és a lépésközt. (Pl. kezdőértéknek 100-at, lépésközt 10-et írva, a független változó a 100, 110, 120, . . . 180, 190 értékeket veszi fel.) Az értékek megadásánál mértékegységet nem kell beírni, a táblázat már tartalmazza azokat. Minden érték beírását a RETURN billentyű megnyomásával kell lezárni. A program ezután kiszámítja a függő változó értékeit, és a három mennyiség összetartozó értékeit táblázat formájában megjeleníti a képernyőn. A függő változót villogó aláhúzás különbözteti meg a független változótól.

Ezekután három lehetőség közül választhatunk. A HELP billentyű megnyomásával a bevezetőhöz lehet visszamenni. A SHIFT és a CLEAR/HOME billentyűk együttes megnyomásával a menühez téhetünk vissza. Így lehetőség nyílik az egyes mennyiségek szerepének megváltoztatására. A CLEAR/HOME gomb megnyomásával az egyes mennyiségeknek újabb értéket adhatunk anélkül, hogy szerepüket (állandó, független- illetve függő változó) megváltoztatnánk.

Ilyenkor mód van arra is, hogy érték beírása nélkül, a RETURN megnyomásával az előzőleg használt értéket megtartsuk.

A program futása a RUN/STOP billentyű megnyomásával szakítható meg. A funkcióbillentyűk azonban a program leállítása után is átprogramozott állapotban maradnak. Ezt a RESET gomb megnyomásával lehet megszüntetni, de ez egyúttal a program törlését is jelenti.

10 rem elektromos teljesitmeny (tabl.2.)

20 rem zatonyi sandor bekescsaba 1986.

60 scnc1r

160 fori=1to7:keyi,"":next:key8,chr\$(140)

170 f1\$=" TELJESITMENY FESZULTSEG ARAMEROSSEG"

180 f2\$=" (W) (V) (A)"

190 v1\$="TELJESITMENY"

200 v2\$="FESZULTSEG"

210 v3\$="ARAMEROSSEG"

220 v\$="-----"

230 sp\$=" "

240 col=9:gosub1990

250 printchr\$(14);chr\$(8);

260 print" q E P ELEKTROMOS TELJESITMENY"

270 print" P=U\*I"

280 print" (tabl.2.)

285 print" "

286 print" -----"

290 print" A program a P = U \* I osszefugges elem-

300 print"zesehez keszult."

310 print" A harom mennyiseg kozul egyet allando"

320 print"ertekunek valasztva egy masik mennyiseg"

330 print"erteke szabadon valtoztathato."

340 print" A program ezek figyelembevetelel"

350 print"kiszamitja a harmadik valtozo ertekeit."

370 print" A CLEAR/HOME megnyomasara uj ertekek"

380 print"adhatok az egyes mennyisegeknek, a"

390 print"SHIFT es CLEAR/HOME egyuttles lenyomasa"

400 print"utan pedig az egyes mennyisegek szerepe"

410 print"valtoztathato meg."

420 char1,25,24,"TOVABB = T"

430 getkeya\$:ifa\$=chr\$(140)thenrun

440 ifa\$<>"t"anda\$<>"T"then430

450 col=10:gosub1990

460 all=50:kezd=10:lep=1

470 print" "+sp\$+ " A VIZSGALT ESET KIVALASZTASA "+sp\$

480 char1,7,5,"FUGGO FUGGETLEN "

490 char1,18,6,"ALLANDO"

500 char1,6,7,"VALTOZO VALTOZO "

510 char1,0,8,v\$

520 char1,0,9," 1. TELJESITM. FESZULTSEG ARAMEROSSEG "+sp\$

530 char1,0,11," 2. TELJESITM. ARAMEROSSEG FESZULTSEG "+v\$

540 char1,0,13," 3. FESZULTSEG TELJESITM. ARAMEROSSEG "+sp\$

550 char1,0,15," 4. FESZULTSEG ARAMEROSSEG TELJESITM. "+v\$

```

560 char1,0,17," 5. ARAMEROSSEG TELJESITM. FESZULTSEG "+sp$
570 char1,0,19," 6. ARAMEROSSEG FESZULTSEG TELJESITM. "+v$
580 char1,0,23," A MEGFELELO SZAM LENYOMASAVAL LEHET VALASZTA
NI."
590 getkey$:ift$=chr$(140)thenrun
600 ift$<"1"ort$>"6"then590
610 col=15:gosub1990
620 print"☐";sp$;" ERTEKADAS " ;sp$
630 char1,2,5,"FUGGO VALTOZO : "
640 char1,2,7,"Allando mennyiseg : "
650 char1,2,9,"Fuggetlen valtozo : "
660 onval(t$)goto670,850,1030,1210,1390,1570
670 :
680 char1,22,5,v1$
690 char1,22,7,v2$
700 char1,22,9,v3$
710 char1,0,11,v$
720 char1,0,14,"A feszultseg erteke (V) : "
730 a=all:p=27:q=14:gosub1750:all=a
740 char1,0,17,"Az aramosseg
750 char1,5,19,"- kezdoerteke (A) : "
760 a=kezd:p=25:q=19:gosub1750:kezd=a
770 char1,5,21,"- lepeskoz (A) : "
780 a=lep:p=22:q=21:gosub1750:lep=a
790 col=9:gosub1990
800 char1,1,1,"☐—————☐"
810 forj=0to9
820 y(j)=all:z(j)=kezd+j*lep:x(j)=all*z(j)
830 next
840 goto1880
850 :
860 char1,22,5,v1$
870 char1,22,7,v3$
880 char1,22,9,v2$
890 char1,0,11,v$
900 char1,0,14,"Az aramosseg erteke (A) : "
910 a=all:p=28:q=14:gosub1750:all=a
920 char1,0,17,"A feszultseg"
930 char1,5,19,"- kezdoerteke (V) : "
940 a=kezd:p=25:q=19:gosub1750:kezd=a
950 char1,5,21,"- lepeskoz (V) : "
960 a=lep:p=22:q=21:gosub1750:lep=a
970 col=9:gosub1990
980 char1,1,1,"☐—————☐"
990 forj=0to9
1000 z(j)=all:y(j)=kezd+j*lep:x(j)=all*y(j)
1010 next
1020 goto1880
1030 :
1040 char1,22,5,v2$
1050 char1,22,7,v1$
1060 char1,22,9,v3$

```



```

1070 char1,0,11,v$
1080 char1,0,14,"A teljesitmeny erteke (W) : "
1090 a=all:p=28:q=14:gosub1750:all=a
1100 char1,0,17,"Az aramosseg"
1110 char1,5,19,"- kezdoerteke (A) : "
1120 a=kezd:p=25:q=19:gosub1750:kezd=a
1130 char1,5,21,"- lepeskoz (A) : "
1140 a=lep:p=22:q=21:gosub1750:lep=a
1150 col=9:gosub1990
1160 char1,15,1," ────"
1170 forj=0to9
1180 x(j)=all:z(j)=kezd+j*lep:y(j)=all/z(j)
1190 next
1200 goto1880
1210 :
1220 char1,22,5,v2$
1230 char1,22,7,v3$
1240 char1,22,9,v1$
1250 char1,0,11,v$
1260 char1,0,14,"Az aramosseg erteke (A) : "
1270 a=all:p=28:q=14:gosub1750:all=a
1280 char1,0,17,"A teljesitmeny"
1290 char1,5,19,"- kezdoerteke (W) : "
1300 a=kezd:p=25:q=19:gosub1750:kezd=a
1310 char1,5,21,"- lepeskoz (W) : "
1320 a=lep:p=22:q=21:gosub1750:lep=a
1330 col=9:gosub1990
1340 char1,15,1," ────"
1350 forj=0to9
1360 z(j)=all:x(j)=kezd+j*lep:y(j)=x(j)/all
1370 next
1380 goto1880
1390 :
1400 char1,22,5,v3$
1410 char1,22,7,v1$
1420 char1,22,9,v2$
1430 char1,0,11,v$
1440 char1,0,14,"A teljesitmeny erteke (W) : "
1450 a=all:p=28:q=14:gosub1750:all=a
1460 char1,0,17,"A feszultseg"
1470 char1,5,19,"- kezdoerteke (V) : "
1480 a=kezd:p=25:q=19:gosub1750:kezd=a
1490 char1,5,21,"- lepeskoz (V) : "
1500 a=lep:p=22:q=21:gosub1750:lep=a
1510 col=9:gosub1990
1520 char1,27,1," ────"
1530 forj=0to9
1540 x(j)=all:y(j)=kezd+j*lep:z(j)=all/y(j)
1550 next
1560 goto1880
1570 :
1580 char1,22,5,v3$
1590 char1,22,7,v2$

```

```

1600 char1,22,9,v1$
1610 char1,0,11,v$
1620 char1,0,14,"A feszultseg erteke (V) : "
1630 a=all:p=26:q=14:gosub1750:all=a
1640 char1,0,17,"A teljesitmeny"
1650 char1,5,19,"- kezdoerteke (W) : "
1660 a=kezd:p=25:q=19:gosub1750:kezd=a
1670 char1,5,21,"- lepeskoz (W) : "
1680 a=lep:p=22:q=21:gosub1750:lep=a
1690 col=9:gosub1990
1700 char1,27,1," ┌──────────────────┐"
1710 forj=0to9
1720 y(j)=all:x(j)=kezd+j*lep:z(j)=x(j)/all
1730 next
1740 goto1880
1750 :
1760 b$="":char1,p,q," "
1770 print" ←┌───┐";getkeya$
1780 ifa$=chr$(13)then1820
1790 ifa$="."or(a$<="9"anda$>="0")thenprinta$;:b$=b$+a$
1800 ifa$=chr$(20)andb$>" "thenprinta$;:b$=left$(b$,len(b$)-1)
1810 iflen(b$)<6then1770
1820 ifb$=""then1850
1830 ifval(b$)<0.01thenchar1,p,q,sp$:goto1750
1840 a=val(b$)
1850 char1,p,q,str$(a)+sp$
1860 ifa<1thenchar1,p,q,"0"
1870 return
1880 :
1890 char1,0,0,f1$:char1,0,3,f2$
1900 forj=0to9
1910 char1,0,6+2*j," "
1920 printusing"#####.#";x(j);
1930 printusing"#####.#";y(j);
1940 printusing"#####.#";z(j);
1950 next
1960 getkeya$:ifa$="┐"then610:elseifa$=""then450:elseifa$=chr$(14
0)thenrun
1970 goto1960
1980 stop
1990 :
2000 scncrlr
2010 color0,col,6:color1,col,1:color4,col,6
2020 return

```

	osztály	csoport	egyéni
új ismeret	●	●	●
alkalmazás			
ellenőrzés			

Betöltési idő: 70 s (28 mp.)

Speciális karakter: nincs

A programmal könnyen vezérelhető metronómmá alakítható a számítógép. A metronóm hang- és fényjelzést is adhat. A program elindítása után rövid tájékoztató jelenik meg a képernyőn. Ezt elolvasva a T gomb megnyomásával léphetünk tovább. Ekkor a METRONOM felirat, alatta az alaphelyzetnek megfelelő periódusidőre (1 s) utaló kiírás, legalul pedig a HANG : KI és a FÉNY : BE üzenetek jelennek meg. A METRONOM felirat alatt a beállított periódusidőnek megfelelő ütemben ad villogó jelzést a program.

A periódusidő a + és - gombokkal 0,1 másodperces, a ↑ és ↓ gombokkal 0,01 másodperces lépésekben növelhető, illetve csökkenthető. A H megnyomásával a hang, az F megnyomásával a fényjelzés kapcsolható ki, illetve be. A pillanatnyi állapotot az utolsó sorban a program folyamatosan feltünteti. Ugyancsak kiírja a beállított periódusidő értékét is.

A CLEAR/HOME billentyű lenyomásával az 1 másodperces idő állítható vissza. Ha a CLEAR/HOME gombot a SHIFT gombbal együtt nyomjuk le, az 1 másodperc beállításával együtt a hang a kikapcsolt, a fény bekapcsolt állapotba kerül. A HELP billentyű megnyomásával a program a tájékoztatóhoz tér vissza.

A program a RUN/STOP lenyomásával állítható le. A funkció-billentyűk eredeti állapota a RESET megnyomásával állítható vissza, de ez a programot is törli a memóriából.

10 rem metronom

20 rem zatonyi sandor bekescsaba 1986.

30 fori=1to7:key1,"":next:key8,chr\$(8)

40 graphic0,1:color0,2,7:color1,1:color4,6,3:printchr\$(14);chr\$(8)

50 print"        ——— ——— ———"

60 print"        | | | | | | | |"

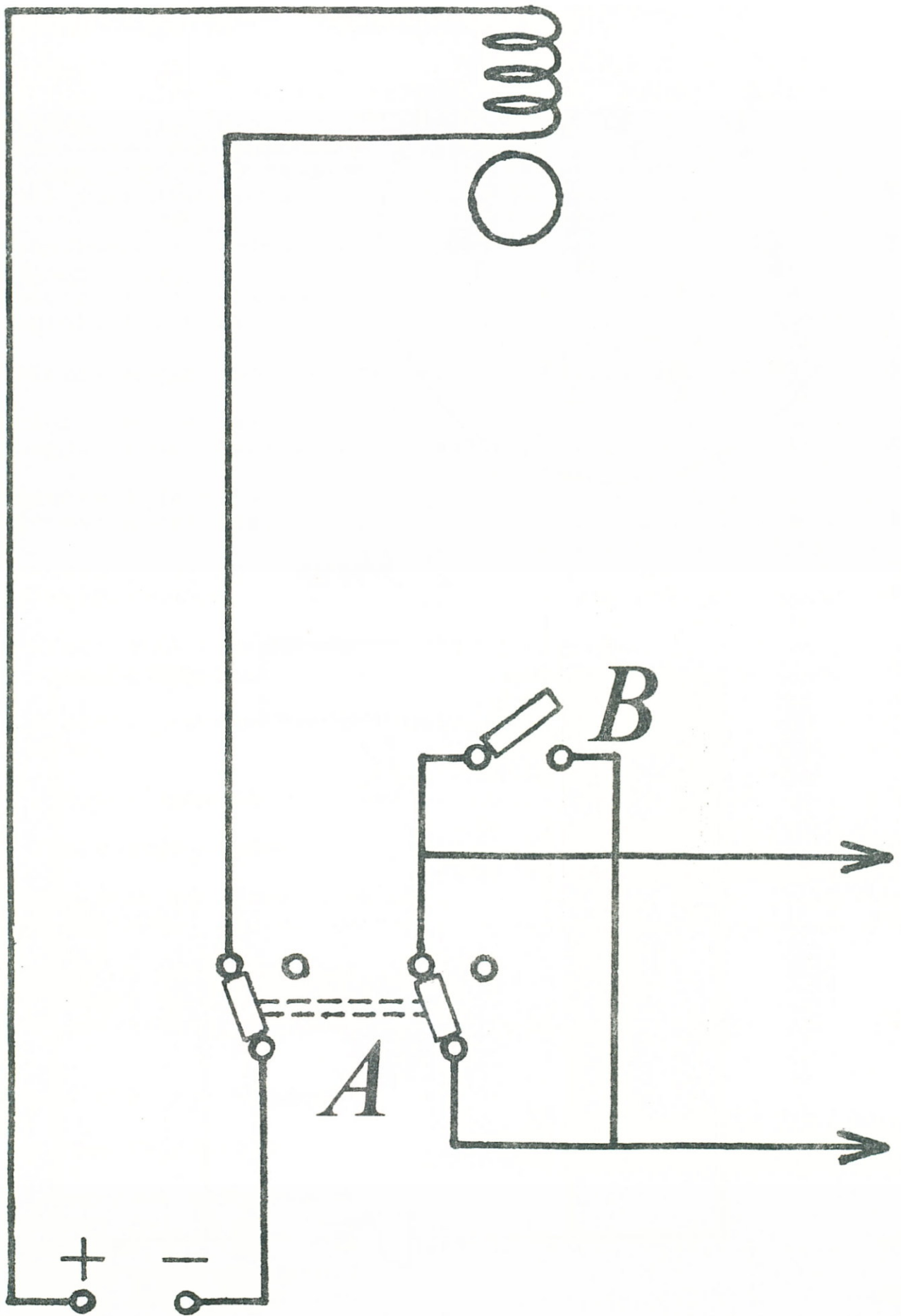
70 print"        | | | | | | | |        M E T R O N O M"

80 print"        o o o o o o o o"

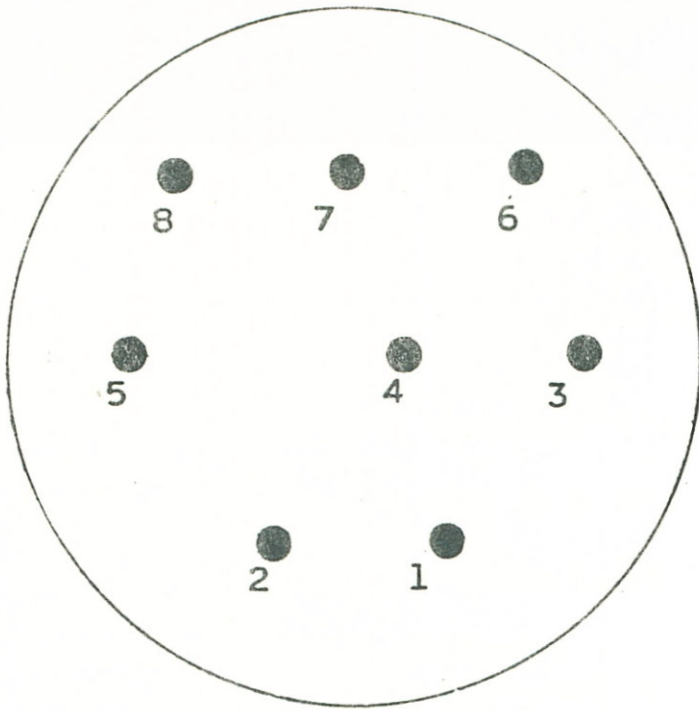
```

90 print"☐☐☐ A programmal egy feny- es hangjelzest      ado metronom
szimulalható."
100 print"☐ A fenyjelzes az F gombbal, a hangjel a H gommbbal ki
/be kapcsolható."
110 print"☐ A metronomon beallithato idotartam      0.3 es 5 maso
dperc kozott lehet."
120 print"☐ A felfele illetve lefele mutato nyilak"
130 print" 0.01 masodperccel, a '+' illetve - gom-"
140 print" bok 0.1 masodperccel valtoztatjak meg      a beallitott i
dot."
150 char1,25,23,"TOVABB = ☐T☐"
160 getkeya$:ifa$<>"t"anda$<>"T"then160
170 scnc1r
180 char1,13,5,"M E T R O N O M"
190 t=100:h=0:f=1
200 char1,10,20,"HANG:" :char1,25,20,"FENY:"
210 goto360
220 fori=0to(t-10)/.131:next
230 char1,13,7,jel$:sound1,800,5
240 fori=0to30:next
250 char1,13,7,"☐ "
260 geta$:ifa$=""then220
270 ifa$=chr$(8)thenrun
280 ifa$="☐"thent=100
290 ifa$="☐"then170
300 ifa$="☐"andt<=499thent=t+1
310 ifa$="☐"andt>=31thent=t-1
320 ifa$="+"andt<=490thent=t+10
330 ifa$="-"andt>=40thent=t-10
340 ifa$="h"thenh=h+1and1
350 ifa$="f"thenf=f+1and1
360 char1,16,15,"T =":printusing"##.## s";t/100
370 ifh=1thenh$="BE":vol7:elseh$="KI":vol0
380 iff=1thenf$="BE":jel$="☐ " :elsef$="KI":jel$="
"
390 char1,16,20,h$:char1,31,20,f$
400 fori=0to100:next:goto260

```

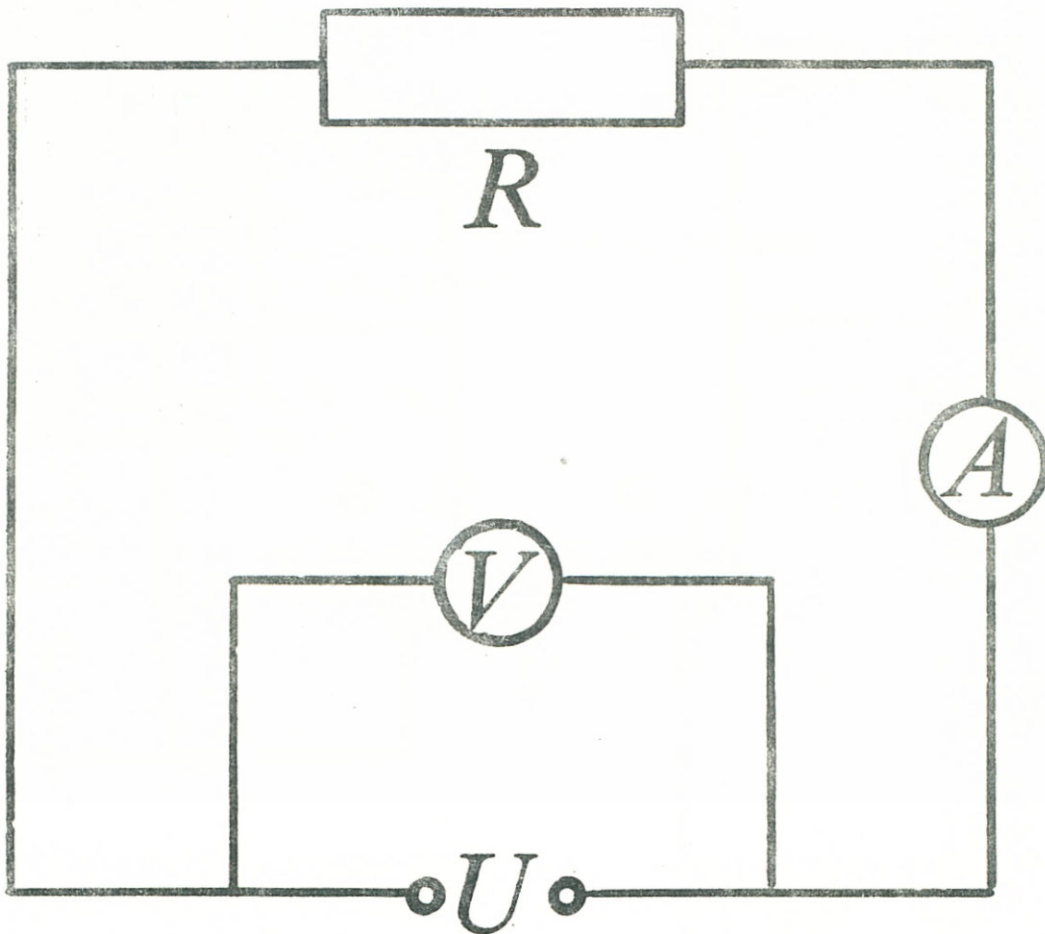


1. ábra



- 1. fel JOY(n)=1
- 2. le JOY(n)=5
- 3. bal JOY(n)=7
- 4. jobb JOY(n)=3
- 5. 5 V
- 6. tűz JOY(n)=128
- 7.
- 8. közös

2. ábra



3. ábra

## TARTALOM

1. A SZÁMÍTÓGÉP ALKALMAZÁSÁNAK METODIKÁJA (Dr. Zátonyi Sándor)	
1.1. A számítógép alkalmazása a tanítási órán és a tanórán kívüli foglalkozásokon .....	3
1.1.1. Számítógép a tanítási órán .....	3
1.1.2. Számítógép a fakultatív foglalkozáson és a szakkörön .....	4
1.1.3. Korrepetálás számítógéppel .....	4
1.2. A számítógép alkalmazása és a "hagyományos" módszerek kapcsolata .....	5
1.3. Módszertani javaslatok a programok alkalmazásához .....	8
1.3.1. Javaslatok az új ismeretek feldolgozásához .....	9
1.3.1.1. Kísérletek bemutatása számítógéppel .....	9
1.3.1.2. A mérési eredmények pontosabbá tétele .....	11
1.3.1.3. A jelenségek szimulálása .....	12
1.3.1.4. Táblázatok elemzése .....	26
1.3.2. Javaslatok az alkalmazáshoz, gyakorláshoz, ellenőrzéshez .....	33
1.4. A programok felhasználási lehetőségei az 1978-as tanterv megvalósításához .....	42
2. PROGRAMOK (Zátonyi Sándor)	
2.1. Általános tudnivalók .....	46
2.1.1. A programok megírásának szempontjai. A programok futtatása .....	46
2.1.2. A botkormány-csatlakozó felhasználása mérésekhez .....	48
2.2. Kezelési útmutatók és programlisták .....	49

2.2.1. Mechanika .....	50
2.2.2. Hőtan .....	175
2.2.3. Fénytan .....	188
2.2.4. Elektromosság .....	198
2.2.5. Egyéb programok .....	233

## **Tisztelt Olvasónk!**

**Felhívjuk figyelmét,** hogy a könyvben ismertetett 46 programot tartalmazó mágneslemez, illetve -kazetta megrendelhető a NOVOTRADE RT. DELTASoft Iroda címén:

1389. Budapest, Pf. 139.

### **A 46 program ára:**

kazettán : 376. – Ft.  
lemezen : 600. – Ft.

( A kazettás változatban a programok gyorsított - turbós - eljárással kerültek felvételre. Erre utal a programismertető részben a betöltési időre vonatkozó - zárójelben közölt - érték.)

NOVOTRADE RT. DELTASoft Iroda



