

C S U P A
C S U P A

J A T É K
S I N C L A I R

Z X S P E C T R U M
n e g y e d i k
r é s z

j á t é k g y á r o s 2 0 0 6



Játékgyáros legjobbjai

NYITRÁISZLET
BASISZLET

program

2005-2006

játékgyáros 2006

A BASIC programozási nyelv

A BASIC programozási nyelv is, mint sokminden más a világon, magyar találmány, egy magyar ember találta ki társával, név szerint Kemény János. Valamint egy bizonyos ember, akit ma a világ leggazdagabbjának is neveznek, köszönheti ennek a programnyelvnek és Kemény Jánosnak azt, amit elért, és ő nem más, mint, Bill Gates.

A BASIC (Beginner's All-purpose Symbolic Instruction Code) programozási nyelvet 1964-ben készítette Kemény János és Thomas Kurtz a Dartmouth College-ben, oktatási céllal. A nyelv az első interpretált nyelv, ami a program futás közben fordítja a gép saját nyelvére. A nyelv alapjait a Fortran programozási nyelv alapján tervezték meg. De menjünk csak szépen sorjában:

A hatvanas évek elején Kemény János és társai azon kezdtek gondolkodni, hogyan használhatnák egyszerre többen a számítógépet. Hiszen amíg a használó gépel vagy a printer nyomtat, a processzor nem csinál semmit! Így fogalmazódott meg az időelosztásos számítógép gondolata. Mindegyik használó saját terminálján saját programjával foglalkozik, a központi számítógép pedig beosztja saját processzorának működési idejét: az minden másodpercet kihasználva végül is mindenkit ellát. Az időelosztás tehát nem a használó, hanem a központi program feladata. A Dartmouth Időelosztásos Rendszer 1963-ban valósult meg. - "Életem egyik legboldogabb pillanata volt, amikor nem kellett többé kártyákat lyukasztgatnom" - emlékezett vissza Kemény. Megnyílt a kapu, hogy sok egyetemi hallgató test(kéz)közelbe juthasson egy számítógéphez. De ehhez a gépi nyelv (vagy akár a FORTRAN) nem volt elég didaktikus. John G. Kemény gondolata volt, hogy ki kell dolgozni a célnak megfelelő interaktív nyelvet; amelynél a gép azonnal reagál a kapott utasításra, így azt a kezdő is gyorsan megtanulhatja próba-szerencse alapon.

Megfogalmazta a kívánalmakat:

- nyelvet a kezdő is könnyen megtanulhassa.
- sokoldalú nyelv legyen:
- bármilyen célra készülhessen program,
- magasszintű utasításai utólag tanulhatók, árát ne a kezdő fizesse, hanem a haladó,
- a nyelv legyen interaktív használó és számítógép között,
- világos, érthető hibaüzeneteket adjon használóinak,
- kis programokra gyorsan válaszoljon,
- használható legyen a gép szerkezetének ismerete nélkül,
- védje a használót a computer operátorrendszerének gondjaitól.

Így született meg John G. Kemény és Tom Kurtz alkotása: a BASIC (**B**eginners' **A**ll-purpose **S**ymbolic **I**nstruction **C**ode), amely a Föld legtöbb ember által értett és használt nyelvei közé tartozik. Az első BASIC program 1964. május 1-én hajnali 4 órakor futott le. Kemény vallja: - "A BASIC nyelvet nem azért találtuk ki, hogy egy újabb számítógépnnyelvet csináljunk. Azért találtam ki, mert úgy éreztem, hogy a számítógépet hozzáférhetővé kell tenni minden egyetemi hallgató számára."- Amikor mindezek kifejlesztésére a Nemzeti Tudományos Alaphoz (ez az amerikai OTKA) pályázatot nyújtott be, bírálói kifogásolták, hogy nem

számítógép-szakértőkkel kíván együttműködni, hanem undergraduate egyetemi hallgatókkal. - "A Tudományos Alap ebben alaposan tévedett. Ma azt mondom, hogy épp azért sikerült elsőként nekünk, mert amíg mások computer-szakértőket használtak, mi egyetemi hallgatókkal dolgoztunk! A hallgatók képesek órákat dolgozni vég nélkül, telve új ideákkal, kreatívan belevágnak a leghetlenebb feladatokba is" - írta Kemény. A BASIC-et ma többen használják, mint az összes többi számítógép-nyelvet együttvéve. 30 esztendővel a BASIC megalkotása után reálisan látjuk: Kemény János álma megvalósult. Valamikor Ford azért alkotta meg a T-modellt, hogy minden polgárnak lehessen autója. Az időelosztás és a BASIC valami hasonlót valósított meg: számítógép programozóvá nevelte a fiatalok millióit. Vannak a magyar földnek olyan szülöttei, akiknek el kellett távozniuk, de eredményeik ma Magyarországon is valósággá váltak: a Szilárd által megálmodott atomenergia, a Wigner által kifejlesztett vízhűtésű reaktor, a Neumann-féle univerzálisan programozható elektronikus számítógép, a Kemény által céltudatosan kifejlesztett számítógép-népnyelv, meg időelosztó operációs rendszer ilyen alkotások. (Időelosztó hálózaton keresztül pesti egyetemi hallgatók is hozzáférhetnek superszámítógéphez.) Tíz esztendővel ezelőtt, 1983-ban a magyar iskoláknak szétszótották az iskolaszámítógépeket a nélkül, hogy a tanárok tudták volna használni őket. A diákok birtokukba vették Neumann és Kemény örökét.

Az IBM első Louis Robinson díját John G. Kemeny nyerte el 1990-ben az időelosztásos rendszer bevezetéséért. Amikor a BASIC megszületett, Kemény és Kurtz a márkanévet levédte, de a nyelvet bárki díjazás nélkül használhatta.

Ez nyilván hozzájárult a BASIC gyors elterjedéséhez. Az első személyi számítógépek ezt választották anyanyelvüknek. Összes memóriakapacitásuk eleinte csak 8 k vagy 16 k volt. Hogy ebbe még egy használóbarát nyelvet is begyömöszölhessenek, a BASIC nagyon karcsúsított változatát kellett alkalmazni. Ez még nem lett volna baj. Később azonban a személyi számítógépek 64 k, majd 640 k és még nagyobb RAM és ROM fölött rendelkeztek, de még ma is lényegében az amputált BASICet használják. Ezért computer-szakértők lenézik a GOTO-val teli BASIC programokat, csak a PASCALt és más strukturált nyelveket ítélik kultúremberhez méltónak. Kemény János azonban hangoztatta, hogy az eredeti DARTMOUTH-BASIC strukturált nyelv volt, csak a mikroszámítógépek gyártói csonkították meg. Ezért 1984-ben Tom Kurtzcal útjára bocsátotta a TRUEBASICet, amely az eredeti DARTMOUTH-BASIC-nek PC-re adaptált modern örököse, teljesen strukturált nyelv, amely futhat interaktív és kompilált módban egyaránt, utasítás-számozás és GOTO nélkül, ugyanakkor érti az elterjedt miniBASICet is. A TRUEBASIC nem vált egyeduralmódóvá, viszont az egymással versenyző vállalatokat hasonló strukturált BASIC-nyelvjárások kidolgozására készítette. - "A TRUEBASIC úgy működött, mint egy csípős légy: a többi céget is arra ösztökölte, hogy értelmes BASICkel rukkoljanak ki." - A TRUEBASIC, TURBOBASIC, QUICKBASIC, VISUAL BASIC fokozatosságot és egyszerűséget megőrző voltát ma már a vajtfülű szakértők is elismerik. Nagy lökést jelentett a nyelv elterjedésében a DOS operációs rendszer elterjedése is, melynek újabb verzióiba beépítették a Qbasic nevű BASIC változatot, melynek a 4.5-ös és a 7.0-ás verziója alkalmas *.exe és *.com kiterjesztésű futtatható program létrehozására is.

Akkor most térjünk rá arra a részre, amit az elején említettem, hogy hogyan is tette Bill Gates-t a BASIC nyelv a világ leggazdagabb emberévé, vagy legalábbis nagyban elősegítette a meggazdagodását:

Bill Gates és Paul Allen (a Microsoft alapítói) a '70-es években próbáltak valami különöset, valami különböző alkotni az akkori számítógépes programoknál, és programnyelveknél. Amikor a M.I.T.S. gyártmányú Altair számítógép megjelent Allen rávette Gates-t, hogy segítsen neki kifejleszteni rá a BASIC nyelv egy változatát, és felkeresték az ötlettel a gyártót. Amikor a M.I.T.S. érdeklődést mutatott a dolog iránt a BASIC nyelv és a PC jövője elkezdődött. Mindeközben Gates a Harvard hallgatója volt, Allen pedig a Honeywellnél dolgozott. Allen és Gates levédették a M.I.T.S. számára az Altair gépekre írt BASIC változatot. Ez a verzió összesen 4K memóriát használt a forráskód és a programkód tárolására. Gates és Allen számos platformra elkészítette a saját BASIC nyelvét. Ebben az időben a Microsoft Corporation egyeduralmódóvá vált a PC-világban. A '70-es évek második felében a BASIC nyelvet alkalmassá tették, hogy Apple, Commodore és Atari gépeken is használható legyen, és ekkor érkezett az idő mikor Bill Gates kiadta a DOS-t, a beépített BASIC értelmezővel. Az IBM-DOS verziója ennek az interpreternek a BASICA volt, melyet az IBM gépek BIOS-ában használtak fel. Azt a verziót, amit az MS-DOS operációs rendszerekkel terjesztettek GW-BASIC-nek hívták, és bármilyen számítógépen képes volt futni, ami képes volt a DOS futtatására. Gyakorlatilag semmi különbség nem volt a BASIC-A és a GW-BASIC között, amely verziót a DOS és a vele együtt értékesített BASIC verzió megjelenése az IBM ötletét értelmetlenné tette.

A Microsoft rájött, hogy mennyire népszerű is a BASIC interpreter, és elhatározta, hogy kiadja azt a fordítót, amely segítségével a felhasználók az interpreter nélkül képesek futtatni a programjaikat. A QuickBasic volt az a megoldás amiben Microsoft ezt megvalósította. Ezt éveken át forgalmazta, egészen a 4.5 verzió megjelenéséig. Ekkor a Microsoft elhatározta, hogy előáll egy még ütősebb termékkel, és elkezdtek terjeszteni a PDS BASIC-et (Professional Development System, Professzionális fejlesztő rendszer) mely pályafutása a 7.1-es verzió megjelenésével ért véget (Ezt hívták még QuickBasic Extendednek). A PDS egy rövid életű ötlet volt, és nem voltak benn igazi lehetőségek (habár gyakorlatilag a QB 4.5 továbbfejlesztése volt). A Microsoft nagy fába vágta a fejszét, mikor megalkotta a BASIC-hez a GUI-t (Graphic User interface, Grafikus felhasználói felület) és létrejött a Visual Basic, mind DOS, mind pedig Windows alá. A DOS verzió az 1.0 verzióval ki is fulladt, bár kiadtak még hozzá egy professzionális frissítést. Az egyetlen különbség a VB for DOS és a QB között csak annyi, hogy a VB lefordítja a QB 4.5 kódot és a professzionális kiadás lefordítja a PDS 7.1 kódot. Még egyetlen egy dolog: a PDS képes OS/2 kódba fordítani a VB-DOS Pro/Std. És a QB 4.5 nem.

Még néhány említést érdemlő, a témához tartozó, tény:

Egy Robert S. Zale nevű egyén felismerte, hogy mekkora lehetőségek is rejlenek a BASIC-ben, és megalkotta a saját fordítóját. A Borland Inc. elhappolta az ötletet és kiadták a TurboBasic-et, de Mr. Zale nemsokkal később kiadta a saját termékét, amit PowerBasicnek nevezett el és egészen a 3.1-es verzióig jutott. A PowerBasic az egyik leghatalmasabb fordító a piacon, melyben megvalósul az előjel nélküli változók ötlete, a belső Assembly kódok használata és még sok hasonlóan szép kiegészítése a BASIC nyelvnek. PB Inc. még kiadta a PowerBasic OS/2-es és Windowsos verzióit, de soha nem hagyta magára a DOS felhasználókat sem. Aki magáról a BASIC nyelvről akar többet megtudni, látogasson el a következő oldalra:

A BASIC nyelv változatai:

Zárójelben a programfuttatási környezet található.

Altair BASIC (MITS Altair, S-100; Microsoft első terméke)
Amiga BASIC (Commodore Amiga)
AMOS BASIC (Commodore Amiga)
Apple Business BASIC (Apple ///)
Applesoft BASIC (Apple II)
ASIC programozási nyelv (MS-DOS)
Atari 2600 Basic Programming (Atari 2600 video játék konzol)
Atari BASIC (aka Sheperdson BASIC) (Atari 8-bites család)
Atari Microsoft BASIC II (Atari 8-bites család)
B32 Business Basic (Data General Eclipse MV, UNIX, MS-DOS)
BASIC09 (OS-9 és OS-9 68K, a Motorola 6809 és 68K processzoron, külön külön)
Microsoft BASICA interpreter (Advanced BASIC, IBM PC korai változatán ROM-ba égetve) (Lemizzel ellátott gépeken a PC-DOS része)
BASIC A Plus+ (Atari 8-bit család)
BASIC-E programnyelv (aka 'submarine BASIC') (CP/M)
Basic Plus (Digital Equipment Corporation gyártmány a RSTS/E OS, PDP-11 processzoron)
BASIC XE programnyelv (Atari 130XE)
BBC BASIC (Acorn/BBC Micro, RISC OS, Tiki 100, Cambridge Z88, CP/M, MS-DOS)
Blitz BASIC (Win32, Commodore Amiga)
Business Basic
Bxbasm (Win32, open source, native compile)
Bywater BASIC (aka bwBASIC) (MS-DOS, POSIX)
Caché Basic (Caché Database egyik parancsleíró nyelve a kettőből)
CBASIC programnyelv (BASIC-E leszármazottja) (CP/M, MS-DOS)
Chipmunk Basic programnyelv (Apple Macintosh, CLI port Win32 részére, GNU/Linux; szabad szoftver)
Color BASIC programnyelv (Tandy Radio Shack TRS-80 Color Computer, aka CoCo)
Commodore BASIC (Commodore 8-bites család)
DarkBASIC & DarkBASIC Professional
Dartmouth BASIC (lásd még: True BASIC)
Data General Business Basic (Data General Nova és későbbi DG miniszámítógépek)
Ethos Game Basic (MS Windows)
Extended Color BASIC (TRS-80 Color Computer)
FutureBASIC (Apple Macintosh)
Gambas (GNU/Linux) (Visual Basic-hez hasonló)
geoBASIC (GEOS Commodore 64 számítógépen)
GeoBASIC (Leica) (Leica TPS 1000/1100 vizsgálati állomásokon)
Gnome Basic (GNU/Linux) (Visual Basic klón)
GFA BASIC (Commodore Amiga, Atari ST)
Microsoft GW-BASIC interpreter (MS-DOS) (BASICA kompatibilis, IBM ROM rutinoktól független)
Hbasic (GNU/Linux) (Visual Basic-hez hasonló)
HotPaw Basic programnyelv (aka yBasic, nee cbasPad Pro) (PalmOS)
IBasic (MS Windows)
Integer BASIC (Apple II)
IS BASIC (Zilog Z80)
Liberty BASIC programnyelv (MS Windows)
Locomotive BASIC (Amstrad CPC)
LotusScript (Lotus Notes)
MAI Basic Four Business Basic (többféle miniszámítógép)

Mallard BASIC programnyelv (Amstrad PCW, CP/M on ZX Spectrum +3)
Microsoft BASIC (Microsoft BASIC variációk) (különbéle
mikroszámítógépeken)
Microsoft MBASIC programnyelv (CP/M operációs rendszer alatt)
MSX BASIC (MSX)
Northstar BASIC (Processor Technology, Northstar, később átdolgozva x86
platformra Basic '86 néven)
NSBasic (Windows CE rendszer alatt)
Phoenix Object Basic programnyelv (GNU/Linux)
PowerBASIC (Turbo BASIC leszármazottja) (MSDOS, Win32)
PureBasic (Win32, GNU/Linux és Commodore Amiga)
QBASIC programnyelv (MS-DOS)
Microsoft QuickBASIC fordító (MS-DOS)
RapidQ (keresztplatform, szabad szoftver)
REALbasic (Apple Macintosh, Win32, Linux)
Revelation BASIC programnyelv (MS-DOS)
SAM Basic programnyelv (Sam Coupé)
ScriptBasic (Win32 and GNU/Linux)
Sharp BASIC (Sharp zsebszámítógép)
Sinclair BASIC (ZX80, ZX81/TS1000, ZX Spectrum)
SmallBASIC (GNU/Linux, MS-DOS, Win32, PalmOS; GPL)
SmartBASIC (Coleco Adam)
ST BASIC (Atari ST)
StarOffice Basic (aka StarBasic) (OpenOffice, StarOffice)
STOS BASIC (Atari ST)
SuperBasic (Sinclair QL)
THEOS Multi-User Basic
TI BASIC programnyelv (Texas Instruments programozható kalkulátorok)
TI BASIC (TI 99/4A) (Texas Instruments TI 99/4A)
TI Extended BASIC programnyelv (ditto)
Tiny BASIC (korai S-100 gépek)
TRS-80 Level I BASIC programnyelv (TRS-80) (TinyBASIC alapú)
TRS-80 Level II BASIC programnyelv (TRS-80)
True BASIC (MS-DOS, MS Windows, Apple Macintosh)
Turbo BASIC (BASIC/Z utód)
VBScript (MS ASP, MS Windows with WSH)
Visual Basic (MS Windows)
Visual Basic .NET (MS Windows)
Visual Basic for Applications (VBA) (MS Office beépített nyelv MS Windows
és Apple Macintosh alapú gépeken)
Watcom Basic
WordBasic (MS Office verziók elődje)
wxBasic (BASIC interpreter wxWidget részére)
XBasic (Win32, GNU/Linux) (GPL)
Yabasic interpreter (Win32, GNU/Linux, Playstation 2) (GPL)

SUDOKU bevezető.

Kilencszer kilenc, olyan mint egy számvarázs..az alapötlet rendkívül egyszerű:

Adott egy kilencszer kilences négyzetrács, háromszor hármaskisebb négyzetekre osztva, a feladat pedig az, hogy minden sorba és oszlopba, illetve a kisebb négyzetekbe írjuk be egytől kilencig a számokat úgy, hogy minden sorban, minden oszlopban és minden háromszor hármaskisebb négyzetben egy-egy számjegyet csak egyszer szerepeljen. (Az, hogy számokat használunk, teljesen érdektelen, lehetnének betűk vagy kis ábrák is, a lényeg, hogy minden sorban, oszlopban és kis négyzetben minden elemből csak egy lehet. Ennek ellenére a 'számos' változat terjedt el leginkább.)

Néhány számot természetesen előre beírnak a feladvány kitalálói, de a rejtvény nehézsége elsősorban nem a megadott számok mennyiségétől függ: könnyen megfejthető lehet egy olyan sudoku, amibe kevés szám van beírva előre, és fordítva. Minden feladvány megfejthető következtetéssel, találgatás nélkül.

A sheffieldi egyetem matematikusa, Frazer Jarvis és Bertram Felgenhauer egyébként kikombinalta, pontosabban szoftverekkel, nyers erővel kiszámolta, hogy a kilencszer kilences rácsban 6 670 903 752 021 072 936 960-féleképpen lehet a szabályoknak megfelelően elhelyezni a számokat.

Egyedülálló - 1-9 - számok

A sudoku egyszerűbb elődjét a 18. század végén alkotta meg Leonhard Euler svájci matematikus, mai formáját először 1979-ben publikálták Number Place (számelhelyezés) néven New Yorkban. Japánban 1984-ben adták ki először, de csak 1986-ban lett hirtelen népszerű. A mostani világméretű közkelettség annak köszönhető, hogy az új-zélandi Wayne Gould, aki korábban bíró volt Hongkongban, 1997-ben Japánban találkozott egy ilyen rejtvényvel, és nekiállt kifejleszteni egy szoftvert, amely képes gyorsan előállítani a rejtvényeket. Amikor kész volt, jelentkezett a brit Times-nál, ahol tavaly november 12-én meg is jelent az első sudoku, ami rövid időn belül sikertörténétté vált.

A sudoku szó "egyedülálló számok"-at jelent japánul, a kifejezés egy japán kiadó birtokában van, emiatt az ország többi kiadója inkább eredeti nevén, *nanpure*-ként jelenti meg a fejtörőket (ez a kifejezés "számelhelyezést" jelent).

Magyarországon azonban nem teljesen ismeretlen a sudoku, ugyanis a Fülesben már valamivel több, mint tíz éve vannak ilyen rejtvények, a bájos „Bűvös négyzet” címmel.

A Sudoku egy japán játék, amely napjainkban Japántól Ausztráliáig szerte a világon százezrek kedvenc időtöltésévé vált. Több száz lap közül naponta Sudoku-rejtvényeket, köztük az olyan tekintélyesek is, mint a The Times vagy a Die Zeit, Magyarországon a Metro.

De mi ez egyáltalán? S hogyan kell játszani?

A Sudoku ugyanis egy olyan 9x9-es négyzetrács, amelyben 9 darab 3x3-as, egyenként 9 darab négyzetet tartalmazó kisebb négyzet van. Összesen tehát 81 darab négyzetből áll. A négyzetrácsban számok vannak megadva és a négyzetrács üres négyzeteibe úgy kell beírunk

a hiányzó számokat, hogy a négyzetrács mind a 9 sorában és mind a 9 oszlopában megtalálható legyen 1-től 9-ig minden egyes szám. De mindezt úgy, hogy ezeknek a feltételeknek a teljesülése mellett még az is igaz legyen, hogy valamennyi (9 darab) kis négyzetben (blokkban) is szerepeljen az összes szám 1-től 9-ig. Ez a rejtvény-megoldás azonban csak akkor érdemli ki a Sudoku elnevezést, ha a nagy négyzetrácsba az előre megadott számok olyan módon vannak megadva, hogy a rejtvénynek csak egyetlenegy megoldása van.

Ugye egész egyszerűnek tűnik?

Megérteni könnyű - megoldani már nehezebb. Szerencsére nem kell matematikusnak lennünk ahhoz, hogy egy ilyen jellegű rejtvényt meg tudjunk fejteni: csak logikus és rendszeres gondolkodásra van hozzá szükségünk. Ám ha úgy érezzük, ez nekünk mégsem megy, akkor sem szabad feladnunk, hiszen a Sudoku épp ezeket - a mindennapi életben és az életbenmaradáshoz oly fontos - képességeket segíthet kifejleszteni. A Sudoku kiváló agytorna: játék a lehetőségekkel, kísérletezés, stratégia.

S mitől függ egy-egy Sudoku nehézsége?

A rejtvények nehézségi szintje attól függ, hogy logikailag mennyire nehéz meghatározni a további számokat. Talán meglepő, de annak, hogy kezdetben hány szám adott a rejtvényben, alig van köze ahhoz, hogy a rejtvény valójában mennyire nehéz. Vannak rejtvények, amelyekben csak kevés számjegy adott, ezek lehetnek akár a legegyszerűbbek is - ugyanakkor lehetséges, hogy azokat a rejtvényeket, amelyekben az átlagnál több szám adott, rendkívül nehéz megfejteni. Ám hogy valójában melyik minősül nehéznek vagy könnyűnek, eléggé szubjektív: nem vagyunk egyformák, másképpen dolgozik az agyunk is.

Mikor játszunk Sudoku-t?

Egy Sudoku megfejtése bármikor kellemes időtöltés lehet: egy hosszú vagy rövid vonatúton, napfényes vagy esős délután, hóban vagy szélben, egyedül vagy nagyobb társaságban. Lehet magányosan gyakorolni, társaságban versenyezni, agyat tornáztatni vele.

A mai korban azonban van még egy segítség, ez a számítógép.

Hát akkor lássunk csodát, hogyan segíthet egy számítógépes program, mely a legegyszerűbb módon, egy BASIC program felhasználásával segít elménknek, a fenti módon összerakni a számokat, függőlegesen, vízszintesen és a 3x3-as kockában elrendezni a számokat 1-től 9-ig.

Számvarázs, avagy SUDOKU-játékprogram, fő részei, a programozás megoldandó lépései:

0-ás sor

Számok és DIM tömbök kiszámítása, beállítása

Alapértékek beállítása

9x9-es tábla megrajzolása

Az első lépés: a tábla, és memória törlése (csak először, kezdetben (!!))

Főmenü elkészítése

Első menü

Második menü

Kész tábla betöltése (egyszerre a memóriában 5 tábla lehet, ez kb. 20K-t foglal a memóriából, a többi maga a program) (Hej-haj a mai gépek sok megabájtja !!)

Tábla mentése (nemcsak emulátoron, hanem az alapgépen is működik) mentés: magnóra, mentés microdrive-re, mentés +3-as géphez floppyra (a mentésekre szükség van, nem biztos, hogy egyszerre meg tudjuk a feladatot oldani)

Játék indítás-harmadik menü

Spectrum tábla-kitöltés segítsége, mutatja a lehetőségeket, vízszintes sorokat, függőleges oszlopokat, és a 3x3-as tömböket.

Javítások a megkezdett táblánál

A táblán való mozgás a spectrum legendás irányítási módja : QAOP SPACE, vagyis Q=fel, A=le, O=balra, P=jobbra, 0-vagy SPACE tűz (beírás) gomb (az emulátort beállíthatjuk a kurzorgomb irányításra)

A befejezéskor a játékgyáros elköszön



```
0 REM Szamvarazs -SUDOKU-program Jatekgyaros
2006 marcius
```

a kedvenc, a BASIC-ban nem is létező 0-ás sor (!!), ezt beírva, az többet nem törölhető, ide szokták a címet, és a szerző nevét beírni, létrehozása az alanti kódsorral lehetséges:

(először a 0-ás sorba írandó szöveget az 1-es számsorral látjuk el, feltesszük a helyére, majd sorszám nélkül), a képernyő aljára írjuk be:

```
POKE PEEK 23635+256*PEEK 23636+1,0
```

majd ENTER-rel elküldjük, és máris látjuk, hogy az 1-es (sor) és sorszám eltűnt, és 0-ra változott.

```
1 DEF FN f(x)=(1 AND (x=1 OR x=3 OR x=5 OR
x=7))
2 DEF FN c(x)=(1 AND (x=2 OR x=3 OR x=6 OR
x=7))
3 DEF FN b(x)=(1 AND (x >= 4 AND x <= 7))
```

Kezdetnek definiáljuk a számokat, ezekre majd a programban hivatkozunk

```
10 BORDER 0: PAPER 4: INK 0
```

a keretet feketére, az alapszínt zöldre, a tinta (betű) színt feketére állítjuk

```

20 DIM s(9,9,10)
30 DIM s$(5,9,9,10,1)
32 DIM r(5)

```

dimenzionáljuk a változók tömbjeit,itt raktározzuk el számokat munka közben,és a program végeztével

(Itt jegyzem meg,hogy a programot soha ne indítsuk RUN-nal,mert akkor a DIM számtömböket lenullázzuk,és minden elvész !!)

```

35 LET elrendez=1
37 LET n$="tabla1"
40 LET alaptabla=0

```

Meghatározzuk a fő változó értékeket,ezek értéke-nagysága kezdetben=0

```
50 BORDER 1: PAPER 1: INK 6: CLS
```

A főcímhöz átállítjuk a táblaszín értékeket :keret fekete,alapszín-háttér kék,rajzoló-író szín sárga.

```

51 FOR f=72 TO 79: POKE 23681,f: LPRINT " S z
a m v a r a z s - j a t e k ": RANDOMIZE USR
65055: NEXT f
52 PRINT INK 5;AT 3,2; PAPER 1;"* NyitraiSOFT-
Jatekgyaros";AT 20,5;"avagy SUDOKU tabla"
54 PAUSE 0

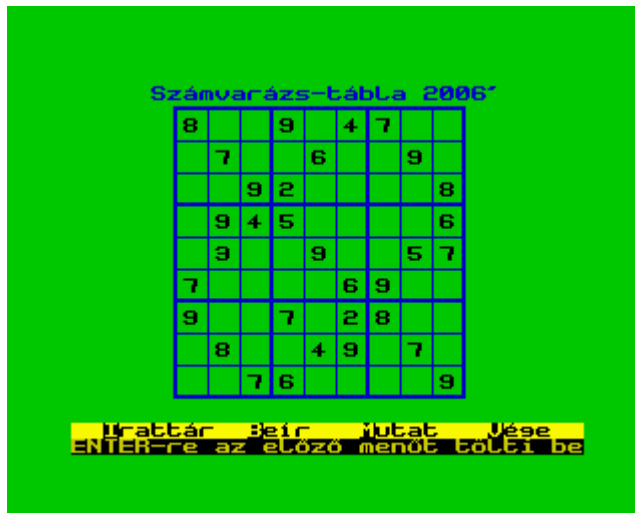
```



a képernyőre középre - sárga színben kb.tízszeres nagyságban kiírja : Számvarázs - játék,a felirat hangjelzés közepette,lüktetve kirajzolódik,felül normál betűkkel' NyitraiSoft-Játékgyáros',alulra 'avagy SUDOKU tábla',majd egy gombnyomásig várakozik..

```
55 GO SUB 9000: GO SUB 9070: GO SUB 9150
```

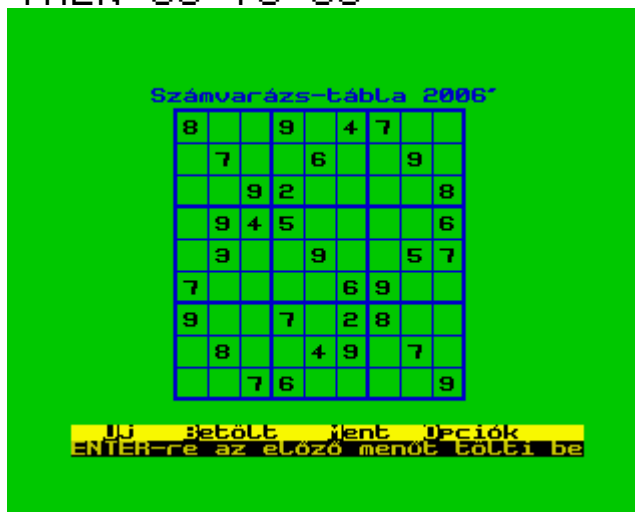
a 9000-es sor írja ki a címet,a 9070-es sor rajzoja ki a táblát,a 9150-s sor visszaküldi ide (részletesebben majd ott)



```

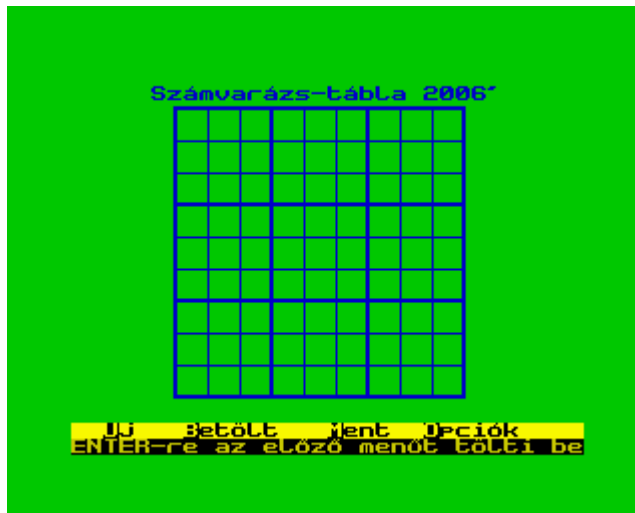
60 LET m$=" Irattar Beir Mutat Vege ":
LET o$="ibmv": GO SUB 8500
70 IF valaszt=0 THEN GO TO 60
    az első főmenű

80 GO TO 100+30*(valaszt-1)
110 LET m$=" Uj Betolt Mentis Opciok
": LET o$="ubmo": GO SUB 8500
115 IF valaszt=0 THEN GO TO 55
  
```



A második főmenű (ezeket használjuk majd legtöbbször)

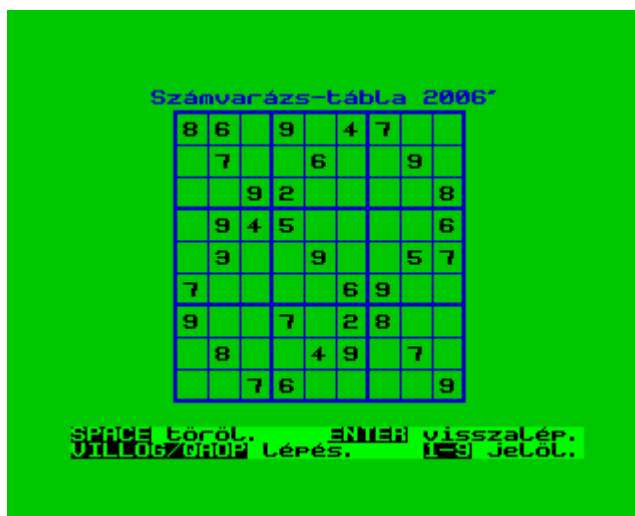
```
120 GO TO 200+50*(valaszt-1)
```



```
130 REM Menü szerkesztő
140 IF alaptábla=0 THEN BEEP .1,-20: GO TO 60
150 GO SUB 9200: GO SUB 7000: GO TO 60
```

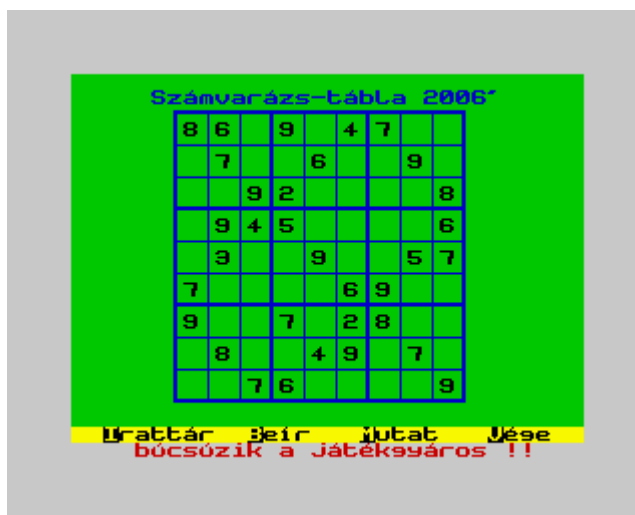
itt eszközölhetjük a javításokat,ha valamit rosszul írtunk be

```
170 IF alaptábla=0 THEN BEEP .1,-20: GO TO 55
180 GO TO 1000
```



a számvarázs-játék indul

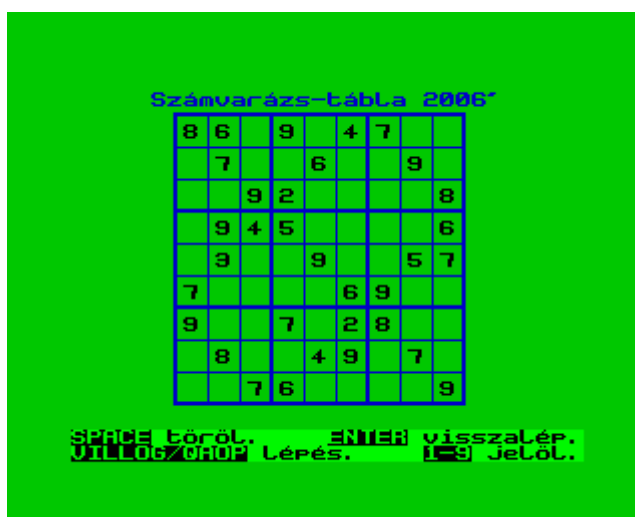
```
192 GO TO 9999
```



itt az utolsó sora a programnak,ha mindent elmentettünk,és már a mai napon nem akarunk a kitöltendő táblával foglalkozni,kiléphetünk a programból..

```
210 GO SUB 9500: GO SUB 9150: LET alaptabla=1:
GO TO 100
260 INPUT "Neve ["; (n$); "]" "; LINE z$
```

az első táblát betölthetjük (magnónál szalagról,emulátornál a megfelelő könyvtárból)



```
265 IF z$="uj" THEN GO TO 100
270 IF LEN z$>10 THEN GO TO 260
280 IF LEN z$>0 THEN LET n$=z$
283 PRINT AT 20,0;
285 IF elrendez=1 THEN LOAD n$ DATA s(): LET
alaptabla=1: GO SUB 9000: GO SUB 9070: GO SUB
9150: GO TO 100
290 IF elrendez=2 THEN LOAD *"m";1;n$ DATA s():
LET alaptabla=1: GO SUB 9000: GO SUB 9070: GO
SUB 9150: GO TO 100
```

```

295 IF elrendez=3 THEN LOAD *"a:"+n$ DATA s():
LET alaptabla=1: GO SUB 9000: GO SUB 9070: GO
SUB 9150: GO TO 100
297 GO TO 100

```

8			4	7	
7			6		9
	9	2			8
	4	5			6
9			9		5
7			6	9	
9			2	8	
8			4		7
	7	6			9

Kimenthetjük más hordozóra is ..

```

310 INPUT "Neve [";(n$);"] "; LINE z$
320 IF LEN z$>10 THEN GO TO 310
330 IF LEN z$>0 THEN LET n$=z$
335 IF elrendez=1 THEN SAVE n$ DATA s(): GO TO
100
340 IF elrendez=2 THEN SAVE *"m";1;n$ DATA s():
GO TO 100
345 IF elrendez=3 THEN SAVE *"a:"+n$ DATA s():
GO TO 100
347 GO TO 100
360 LET m$=" Magnokazettara mentes "
LET o$="m": GO SUB 8500
370 IF választ=0 THEN GO TO 100

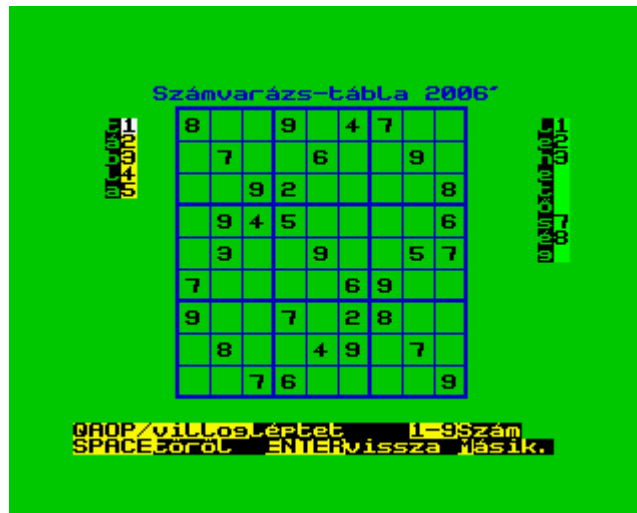
```

ha a tábla a képernyőn van,akkor hozzáláthatunk a feladatnak (persze,ha nem akarjuk,nem kell a számítógép segítségét igénybevenni,nekiláthatunk a képernyőn saját magunk próbálkozása által is kitölteni,megoldani a feladatot.)

```

380 LET elrendez=választ: GO TO 100
1000 REM számvarázs-jatek indul
1010 INPUT ""
1020 PRINT #0; BRIGHT 1; INK 0; PAPER 6;AT 0,0;"
1030 LET cf=1: LET cc=1: LET menu=0
1050 GO SUB 6900
1070 GO SUB 8800

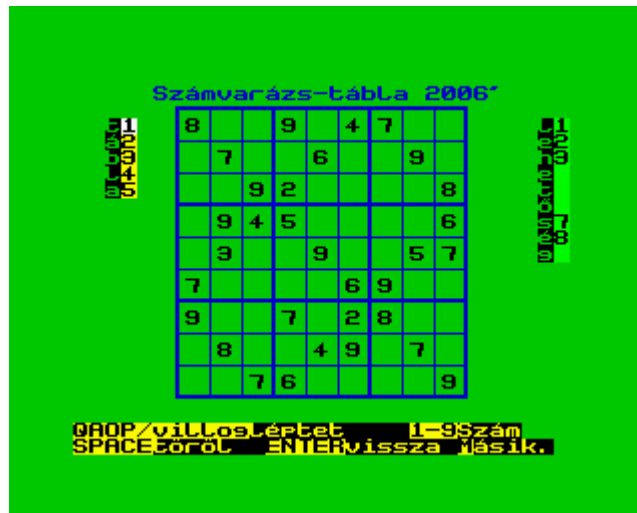
```



Játék indítás-harmadik menü

```

1080 INPUT "": IF menu=0 THEN PRINT #0; BRIGHT
1; INK 6; PAPER 0; AT 0,0;"@ADP/villog leptet
1-9 Szám" "SPACE torol ENTER vissza Masik.": GO
TO 1100
1090 IF menu=1 THEN PRINT #0; BRIGHT 1; INK 6;
PAPER 0; AT 0,0;"Sor Oszlop Tomb Elhataroz" "
Load(tabla) Kiment(tabla) Masik. ": GO TO 1100
1100 REM első kor
1120 LET sor=3: LET oszlop=30: BRIGHT 1: FOR k=1
TO 9: PRINT AT sor+k-1,oszlop;CHR$( (32 AND
s(cf,cc,k)=0)+(48+k AND s(cf,cc,k)<>0)): NEXT k:
BRIGHT 0
1130 LET cfp=3+(cf-1)*2: LET ccp=7+(cc-1)*2
1140 PRINT AT cfp,ccp; OVER 1; INK 8; FLASH 1;"
": BEEP .01,0
1150 PAUSE 0: LET t#=INKEY$
1160 IF t#="o" OR t#=CHR$ 8 THEN PRINT AT
cfp,ccp; OVER 1; INK 8; FLASH 0;" ": LET cc=cc-
1: IF cc<1 THEN LET cc=9
1170 IF t#="p" OR t#=CHR$ 9 THEN PRINT AT
cfp,ccp; OVER 1; INK 8; FLASH 0;" ": LET
cc=cc+1: IF cc>9 THEN LET cc=1
1180 IF t#="a" OR t#=CHR$ 10 THEN PRINT AT
cfp,ccp; OVER 1; INK 8; FLASH 0;" ": LET
cf=cf+1: IF cf>9 THEN LET cf=1
1190 IF t#="q" OR t#=CHR$ 11 THEN PRINT AT
cfp,ccp; OVER 1; INK 8; FLASH 0;" ": LET
cf=cf-1: IF cf<1 THEN LET cf=9
1200 IF t#<"1" OR t#>"9" THEN GO TO 1230
1210 LET ii=cf: LET jj=cc: GO SUB 8600: FOR i=1
TO LEN v$: IF t#=v$(i) THEN LET va=VAL t$: LET
s(cf,cc,10)=va: FOR k=1 TO 9: LET s(cf,cc,k)=(1
AND k=va)+(0 AND k<>va): NEXT k: PRINT AT
cfp,ccp; INK 1;va
1220 NEXT i
1230 IF t#=" " THEN LET s(cf,cc,10)=0: FOR k=1
TO 9: LET s(cf,cc,k)=1: NEXT k: LET i=cf: LET
j=cc: GO SUB 7200: PRINT AT cfp,ccp;" "
1240 IF t#=CHR$ 13 THEN GO TO 50
  
```

a spectrum kérésre megmutatja:milyen számokat írhatunk be a megfelelő sorba,oszlopba,vagy 3x3 tömbbe

```

1250 IF t$="s" THEN PRINT AT cfp,ccp; OVER 1;
INK 8; FLASH 0; " ": GO SUB 2000: GO TO 1030
1260 IF t$="o" THEN LET menu=NOT menu: GO TO
1080
1270 IF t$="t" THEN GO SUB 3000: GO TO 1030: REM
aktualis sor
1280 IF t$="e" THEN GO SUB 4000: GO TO 1030: REM
aktualis oszlop
1290 IF t$="l" THEN GO SUB 5000: GO TO 1030: REM
aktualis 3x3-as blokk
1300 IF t$="k" THEN GO SUB 1500: GO TO 1070
1310 IF t$="m" THEN GO SUB 1600: GO TO 1070
1400 GO TO 1100

```

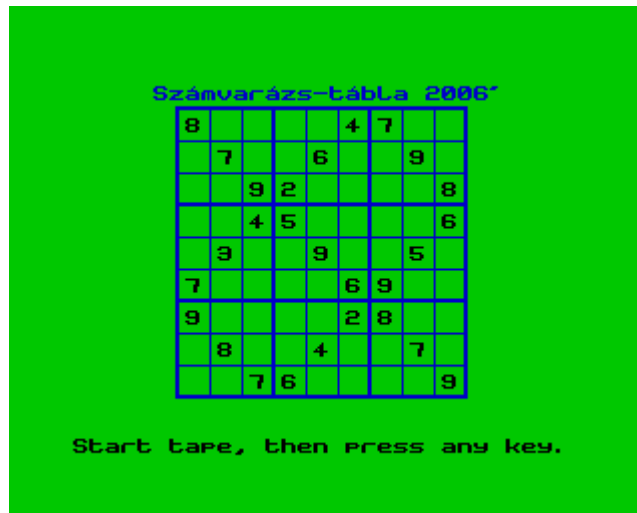
Lehetőség van új tábla kitöltésre,ezt kiválaszthatjuk egy feladványfüzetből,vagy magunk kreálhatunk új táblát..

```

1500 REM tabla kitoltes
1510 INPUT "": PRINT #0;AT 1,0;" 1-5 tabla
mentes,ENTER kilepes ": BEEP .1,20
1520 PAUSE 0: LET t$=INKEY$: IF (t$<"1" OR
t$>"5") AND t$<>CHR$ 13 THEN GO TO 1520
1525 IF t$=CHR$ 13 THEN RETURN
1530 LET res=VAL t$
1540 IF r(res)=0 THEN BEEP .1,-20: GO TO 1520
1550 INPUT "": PRINT #0;AT 1,0;"Betoltom a: ";
1560 FOR i=1 TO 9: PRINT #0;" ";: FOR j=1 TO 9:
FOR k=1 TO 10: LET va=CODE s$(res,i,j,k): LET
s(i,j,k)=va-100: NEXT k: NEXT j: NEXT i
1565 GO SUB 9150
1570 RETURN
1600 REM

```

ha elkészültünk egy feladvány beírásával (ez még nem a kész táblát jelenti,hanem csak az induló számok beírását, mindig mentjük el),hogya később visszatérhessünk az induló álláshoz



```

1610 INPUT "": PRINT #0; AT 1,0; "1-5 tabla
kimentes, ENTER kilépes ": BEEP .1,20
1620 PAUSE 0: LET t$=INKEY$: IF (t$<"1" OR
t$>"5") AND t$<>CHR$ 13 THEN GO TO 1620
1625 IF t$=CHR$ 13 THEN RETURN
1630 LET res=VAL t$
1650 INPUT "": PRINT #0; AT 1,0; "Mentem a tablat
":
1660 FOR i=1 TO 9: PRINT #0; " : "; FOR j=1 TO 9:
FOR k=1 TO 10: LET va=s(i,j,k): LET
s$(res,i,j,k)=CHR$ (va+100): NEXT k: NEXT j:
NEXT i
1670 LET r(res)=1
1680 RETURN

```

itt kezdődik a segítség (persze ne gondoljuk azt, hogy a spectrum egyszerűen kitölti nekünk a feladványt, csak megmutatja, kiírja a lehetőségeket, látványos formában) minden sorban, oszlopban, 3x3-as tömbben megmutatja, mit is írhatunk be.

```

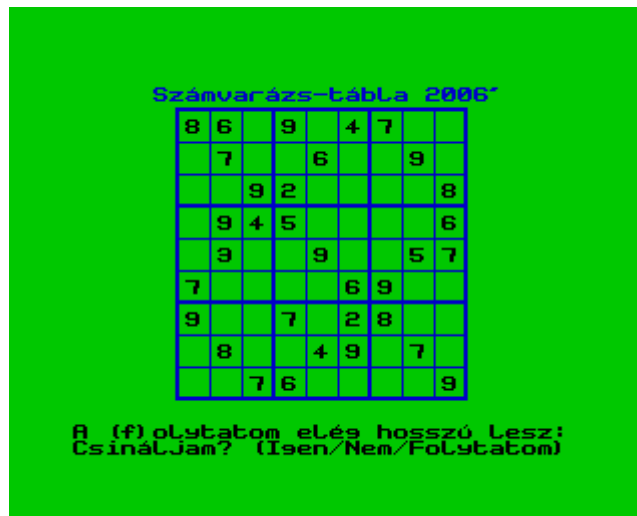
2000 REM Spectrum segíti kitölteni a SUDOKU-
tablat
2010 LET mindig=0: LET elvon=1: LET torol=0
2020 INPUT "": PRINT #0; AT 1,0; INVERSE 1; "A
meghatározó számok keresése "
2025 LET va=0: LET va1=0: LET va2=0
2030 FOR i=1 TO 9: FOR j=1 TO 9: LET cfp=3+(i-
1)*2: LET ccp=7+(j-1)*2: PRINT AT cfp,ccp;
INVERSE 1; OVER 1; " ": IF s(i,j,10)>0 THEN GO TO
2100
2050 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30; CHR$ ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0
2091 IF elvon THEN GO SUB 7200
2092 LET ii=i: LET jj=j: GO SUB 8600
2093 IF LEN v$<>1 THEN GO TO 2100
2094 LET va=VAL v$: LET s(i,j,10)=va: FOR k=1 TO
9: LET s(i,j,k)=(0 AND k<>va)+(1 AND k=va): NEXT
k: GO SUB 7500: PRINT AT cfp,ccp; INK 1; va: GO
TO 2110
2100 PRINT AT cfp,ccp; INVERSE 1; OVER 1; " "
2105 IF INKEY$=" " THEN LET torol=1: RETURN
2110 NEXT j: NEXT i
2120 LET elvon=0
2131 GO SUB 8900: IF határoz THEN RETURN
2132 IF va=0 THEN GO SUB 7900: IF va=0 AND NOT
torol THEN GO SUB 8300: IF va1=0 THEN RETURN
2133 IF torol THEN RETURN

```

```

2134 IF val<>0 AND va=0 THEN GO TO 2020
2135 IF mindig=1 THEN LET t$="i": GO TO 2170

```



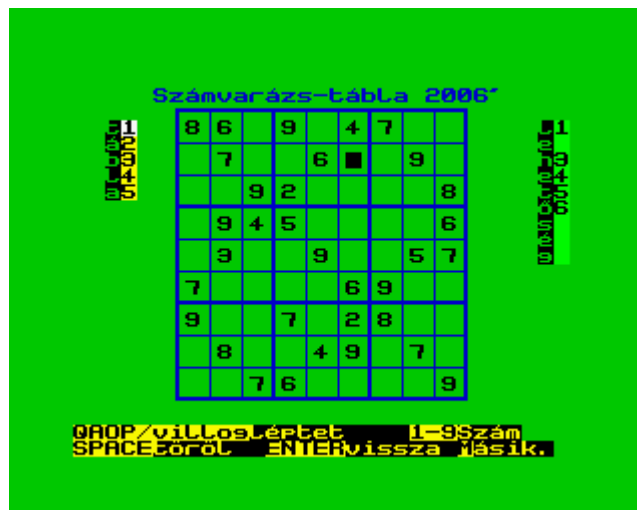
Ha végzett egy-egy résszel, felajánlja a szünetet, vagy a folytatást

```

2140 INPUT "": PRINT #0; AT 0,0; "A (f)olytatom
     elég hosszú lesz: " ^ "CsinálJám?
     (Igen/Nem/Folytatom)"
2150 PAUSE 0: LET t$=INKEY$
2160 IF t$<>"i" AND t$<>"n" AND t$<>"f" THEN GO
     TO 2150
2165 IF t$="f" THEN LET mindig=1: GO TO 2135
2170 INPUT "": PRINT #0; "Már dolgozom is...":
     FOR f=3 TO 19 STEP 2: FOR c=7 TO 23 STEP 2:
     PRINT AT f,c; OVER 1; " ": NEXT c: NEXT f: IF
     t$="i" THEN GO TO 2020
2180 RETURN

```

kérhetünk egy-egy kiválasztott sort



```

3000 REM aktualis sor valasztasa
3010 LET m#="Megmutatom az 1-9 lehetosegeit ":
LET o#="123456789m"
3015 GO SUB 8500
3020 IF valaszt=10 THEN GO TO 3100
3030 IF valaszt<>0 THEN GO TO 3040
3031 FOR j=1 TO 9: IF s(cf,j,10)<1 THEN PRINT AT
cfp,7+(j-1)*2; " "
3032 NEXT j
3035 RETURN
3040 FOR j=1 TO 9: IF s(cf,j,10)<1 AND
s(cf,j,valaszt)=1 THEN PRINT AT cfp,7+(j-1)*2;
INK 3;valaszt: GO TO 3050
3045 IF s(cf,j,10)<1 THEN PRINT AT cfp,7+(j-
1)*2; " "
3050 NEXT j
3060 GO TO 3015

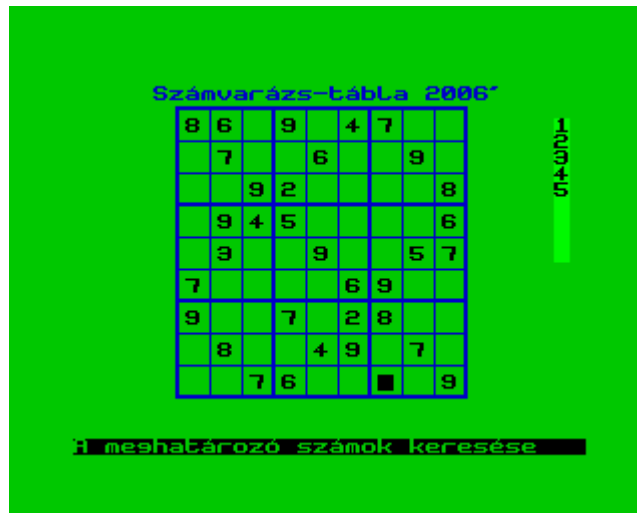
```

megmutatja egy táblázatban,hogyan helyezkednek el a számok,sorban lekérhetjük az 1,2 9 stb. sort is

```

3100 REM aktualis sor reszlete
3120 CLS : LET fi=cf
3125 GO SUB 9000: INK 2: PLOT 52,122: DRAW
144,0: PLOT 52,108: DRAW 144,0: PLOT 52,122:
DRAW 0,-92: DRAW 144,0: PLOT 196,122: DRAW 0,-92
3127 FOR j=1 TO 10: PLOT 52+(j-1)*16,122: DRAW
0,-92: IF j=1 OR j=4 OR j=7 OR j=10 THEN DRAW
1,0: DRAW 0,92
3128 NEXT j
3129 LET v#="Lehetseges": BRIGHT 1: LET sor=7:
LET oszlop=5: GO SUB 8700: BRIGHT 0
3130 PRINT AT 4,13; INVERSE 1;"sor: ";fi
3140 FOR j=1 TO 9: PRINT AT 7,7+(j-1)*2; IF
s(fi,j,10)>=1 THEN PRINT s(fi,j,10): FOR k=1 TO
9: PRINT AT 8+k,7+(j-1)*2; " ": NEXT k: GO TO
3175
3150 PRINT " ": FOR k=1 TO 9: PRINT AT 8+k,7+(j-
1)*2; : IF s(fi,j,k)=1 THEN PRINT k: GO TO 3170
3160 PRINT " "
3170 NEXT k
3175 NEXT j
3180 LET m#=" 1-9 pontos reszletezes "
3190 LET o#="123456789"
3200 GO SUB 8500
3210 IF valaszt=0 THEN CLS : GO SUB 9000: GO SUB
9070: GO SUB 9150: GO SUB 6900: GO SUB 8800: GO
TO 3000
3220 LET fi=valaszt: GO TO 3130
4000 REM aktualis oszlop valasztas
4010 LET m#="Megmutatom az 1-9 lehetosegeit ":
LET o#="123456789m"
4015 GO SUB 8500
4020 IF valaszt=10 THEN GO TO 4100
4030 IF valaszt<>0 THEN GO TO 4040
4031 FOR i=1 TO 9: IF s(i,cc,10)<1 THEN PRINT AT
3+(i-1)*2,ccp; " "
4032 NEXT i
4035 RETURN
4040 FOR i=1 TO 9: IF s(i,cc,10)<1 AND
s(i,cc,valaszt)=1 THEN PRINT AT 3+(i-1)*2,ccp;
INK 3;valaszt: GO TO 4050
4045 IF s(i,cc,10)<1 THEN PRINT AT 3+(i-
1)*2,ccp; " "
4050 NEXT i
4060 GO TO 4015

```



megmutatja egy táblázatban, hogyan helyezkednek el a számok, sorban lekérhetjük az 1,2 9 stb. oszlopot is

```

4100 REM aktualis oszlop
4110 CLS : GO SUB 9000
4112 INK 2: FOR i=0 TO 9: PLOT 52,(3+(i-
1)*2)*8+4: DRAW 160,0: IF i/3=INT (i/3) THEN
PLOT 52,(3+(i-1)*2)*8+5: DRAW 160,0
4114 NEXT i
4116 PLOT 52,12: DRAW 0,18*8+1: PLOT 68,12: DRAW
0,18*8+1: PLOT 52+160,12: DRAW 0,18*8+1: INK 0
4118 LET v$="Lehetséges": LET sor=7: LET
oszlop=29: BRIGHT 1: GO SUB 8700: BRIGHT 0
4120 LET co=cc
4200 FOR i=1 TO 9: PRINT AT 3+(i-1)*2,7;: IF
s(i,co,10)>=1 THEN PRINT s(i,co,10): FOR k=1 TO
9: PRINT AT 3+(i-1)*2,9+(k-1)*2;" ": NEXT k: GO
TO 4220
4205 PRINT " "
4210 FOR k=1 TO 9: PRINT AT 3+(i-1)*2,9+(k-
1)*2;: IF s(i,co,k)=1 THEN PRINT k: GO TO 4215
4212 PRINT " "
4215 NEXT k
4220 NEXT i
4225 INVERSE 1: LET v$="oszlop "+STR$(co): LET
sor=6: LET oszlop=4: GO SUB 8700: INVERSE 0
4230 LET m$="      1-9 lehetséges oszlop "
4240 LET o$="123456789"
4250 GO SUB 8500
4260 IF valaszt=0 THEN CLS : GO SUB 9000: GO SUB
9070: GO SUB 9150: GO SUB 6900: GO SUB 8800: GO
TO 4000
4270 LET co=valaszt: GO TO 4200
5000 REM

```

megmutatja egy táblázatban, hogyan helyezkednek el a 3x3 tömb számai, sorban lekérhetjük az 1,2 9 stb. kockát is

```

5010 LET m$="Megmutatom az 1-9 lehetőségeit ":
LET o$="123456789m"
5020 GO SUB 8500
5030 IF valaszt=10 THEN GO TO 5100
5040 IF valaszt<>0 THEN GO TO 5070
5050 LET ii=1+3*INT ((cf-1)/3): LET jj=1+3*INT
((cc-1)/3): FOR i=ii TO ii+2: FOR j=jj TO jj+2:
IF s(i,j,10)<1 THEN PRINT AT 3+(i-1)*2,7+(j-1)*2;" "

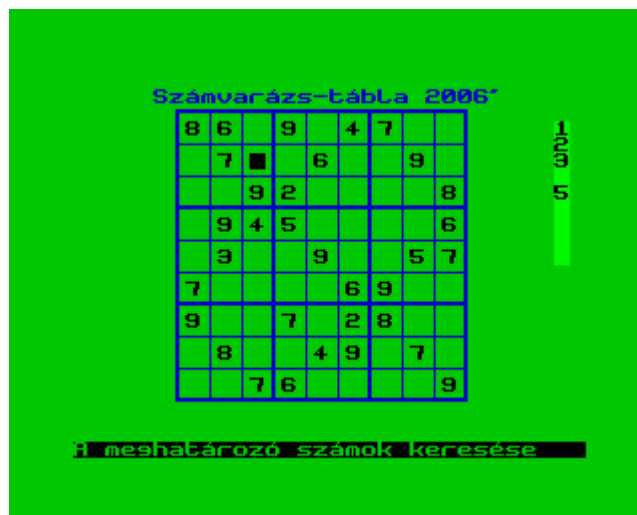
```

```

5060 NEXT j: NEXT i: RETURN
5070 LET ii=1+3*INT ((cf-1)/3): LET jj=1+3*INT ((cc-
1)/3): FOR i=ii TO ii+2: FOR j=jj TO jj+2: IF
s(i,j,10)<1 AND s(i,j,valszt)=1 THEN PRINT AT 3+(i-
1)*2,7+(j-1)*2; INK 3;valszt: GO TO 5080
5075 IF s(i,j,10)<1 THEN PRINT AT 3+(i-1)*2,7+(j-
1)*2; " "
5080 NEXT j: NEXT i
5090 GO TO 5020
5100 REM hangjelzes es ENTER-re kilepes
5110 BEEP 1,-20: GO TO 5020
6900 REM lehetőségek kijelzése
6910 LET sor=3: LET oszlop=2: LET v$="tabla": INVERSE
1: GO SUB 8700
6920 LET sor=3: LET oszlop=29: LET v$="lehetőség": GO
SUB 8700: INVERSE 0
6930 RETURN

```

ha saját magunk írtunk be valamit,akkor a spectrum ellenőriz,és megmutatja a hibás elhelyezkedést



```

7000 REM a spectrum ellenőrzi es megmutatja a
lehetseges szamokat
7005 INPUT "": PRINT #0;AT 1,0; INVERSE
1;"Beallitom a lehetseges szamokat!"
7010 FOR i=1 TO 9: FOR j=1 TO 9: LET cfp=3+(i-
1)*2: LET ccp=7+(j-1)*2: PRINT AT cfp,ccp;
INVERSE 1; OVER 1;" "": IF s(i,j,10)>0 THEN GO TO
7125
7040 GO SUB 7200
7125 PRINT AT cfp,ccp; INVERSE 1; OVER 1;" "":
NEXT j: NEXT i
7140 RETURN

```

vízszintesen,majd függőlegesen is végignézi,és kijelöli a spectrum

```

7200 REM először vízszintesen,majd függőlegesen
is végignézi,es kijelöli
7220 FOR c=1 TO 9: IF c<>j AND s(i,c,10)>0 THEN
LET s(i,j,s(i,c,10))=0
7240 NEXT c
7260 FOR f=1 TO 9: IF f<>i AND s(f,j,10)>0 THEN
LET s(i,j,s(f,j,10))=0

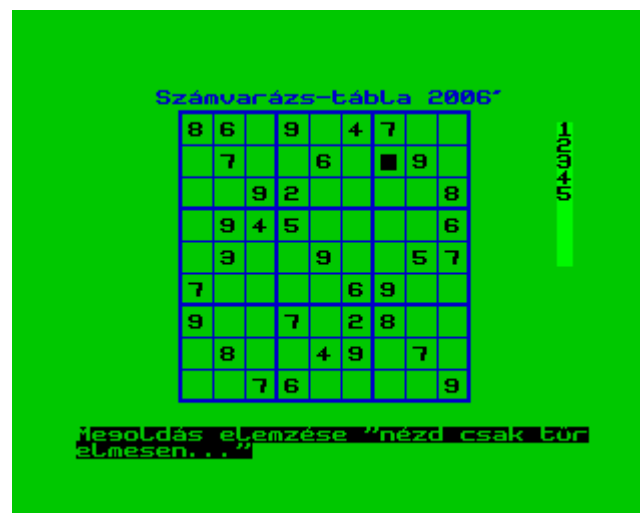
```

```

7280 NEXT f
7290 LET fcom=1+3*INT ((i-1)/3): LET
ccom=1+3*INT ((j-1)/3): FOR f=fcom TO fcom+2:
FOR c=ccom TO ccom+2
7320 IF (f<>i OR c<>j) AND s(f,c,10)>0 THEN LET
s(i,j,s(f,c,10))=0
7330 NEXT c: NEXT f
7340 RETURN
7500 REM a 3x3-as kockakban mutatja a lehetséges
ertekeket
7510 FOR c=1 TO 9: LET s(i,c,va)=(1 AND c=j)+(0
AND c<>j): NEXT c
7520 FOR f=1 TO 9: LET s(f,j,va)=(1 AND f=i)+(0
AND f<>i): NEXT f
7530 LET fcom=1+3*INT ((i-1)/3): LET
ccom=1+3*INT ((j-1)/3): FOR f=fcom TO fcom+2:
FOR c=ccom TO ccom+2: LET s(f,c,va)=(1 AND (f=i
AND c=j))+(0 AND (f<>i OR c<>j)): NEXT c: NEXT f
7540 RETURN

```

Megoldások elemzése ""nézd csak türelmesen..."" ,viszintesen a sort,függőlegesen az oszlopokat,és külön-külön a 3x3 tömböket..



```

7900 REM elemzes minden lehetőséggel
7905 INPUT "": PRINT #0; AT 1,0; INVERSE
1;"Megoldas elemzése ""nezd csak
türelmesen...""
7907 LET va=0
7910 FOR i=1 TO 9: LET cfp=3+(i-1)*2: DIM
v(9,2): FOR j=1 TO 9: LET ccp=7+(j-1)*2: PRINT
AT cfp,ccp; OVER 1; INK 8; INVERSE 1;" "": IF
s(i,j,10)>0 THEN GO TO 7925
7915 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30;CHR$( ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0: FOR k=1 TO 9:
LET v(k,1)=v(k,1)+s(i,j,k): IF v(k,1)=1 AND
v(k,2)=0 THEN LET v(k,2)=j
7920 NEXT k

```

```

7925 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1;" ": IF INKEY$=" " THEN LET torol=1: RETURN
7927 NEXT j: FOR k=1 TO 9: IF v(k,1)=1 THEN LET
s(i,v(k,2),10)=k: LET va=k: LET j=v(k,2): FOR
l=1 TO 9: LET s(i,j,l)=(1 AND l=va)+(0 AND
l<>va): NEXT l: GO SUB 7500: PRINT INK 1;AT
cfp,7+(j-1)*2;va
7930 NEXT k: NEXT i
7935 IF va<>0 THEN RETURN
7950 FOR j=1 TO 9: LET ccp=7+(j-1)*2: DIM
v(9,2): FOR i=1 TO 9: LET cfp=3+(i-1)*2: PRINT
AT cfp,ccp; OVER 1; INK 8; INVERSE 1;" ": IF
s(i,j,10)>0 THEN GO TO 7960
7955 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30;CHR$ ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0: FOR k=1 TO 9:
LET v(k,1)=v(k,1)+s(i,j,k): IF v(k,1)=1 AND
v(k,2)=0 THEN LET v(k,2)=i
7957 NEXT k
7960 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1;" ": IF INKEY$=" " THEN LET torol=1: RETURN
7965 NEXT i: FOR k=1 TO 9: IF v(k,1)=1 THEN LET
s(v(k,2),j,10)=k: LET va=k: LET i=v(k,2): FOR
l=1 TO 9: LET s(i,j,l)=(1 AND l=va)+(0 AND
l<>va): NEXT l: GO SUB 7500: PRINT INK 1;AT
3+(i-1)*2,ccp;va
7970 NEXT k: NEXT j
7980 IF va<>0 THEN RETURN
8000 FOR m=1 TO 9 STEP 3: FOR n=1 TO 9 STEP 3
      a 3x3-as kocka ellenörzése,itt szokott a
      legtöbb hiba lenni

8005 DIM v(9,3)
8010 FOR i=m TO m+2: LET cfp=3+(i-1)*2: FOR j=n
TO n+2: LET ccp=7+(j-1)*2
8020 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1;" ": IF s(i,j,10)>0 THEN GO TO 8050
8030 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30;CHR$ ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0: FOR k=1 TO 9:
LET v(k,1)=v(k,1)+s(i,j,k): IF v(k,1)=1 AND
v(k,2)=0 THEN LET v(k,2)=i: LET v(k,3)=j
8040 NEXT k
8050 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1;" ": IF INKEY$=" " THEN LET torol=1: RETURN
8055 NEXT j: NEXT i
8060 FOR k=1 TO 9: IF v(k,1)=1 THEN LET
i=v(k,2): LET j=v(k,3): LET va=k: LET
s(i,j,10)=va: FOR l=1 TO 9: LET s(i,j,l)=(1 AND
l=va)+(0 AND l<>va): NEXT l: GO SUB 7500: PRINT
INK 1;AT 3+(i-1)*2,7+(j-1)*2;va
8070 NEXT k
8080 NEXT n: NEXT m
8090 RETURN

```

a többi számhoz viszonyított lehetőség-megmutatom a megoldást:

Számvarázs-tábla 2006*

00400000
00400000

8	9	4	7		
7		6		9	
	9	2		8	
9	4	5		6	
3		9		5	7
7			6	9	
9		7	2	8	
	8		4	9	7
		7	6		9

00400000
00400000

005 tábla kimentés, 00000 kilépés

```

00300 REM a többi számhoz viszonyított lehetőség
00300 INPUT "": PRINT #0; AT 1,0; INVERSE 1; "
Megmutatom a megoldást:
00307 LET va1=0: LET va2=0
00310 FOR i=1 TO 9: LET cfp=3+(i-1)*2: LET ii=i:
FOR j=1 TO 9: LET ccp=7+(j-1)*2: PRINT AT
cfp,ccp; OVER 1; INK 8; INVERSE 1; " ": IF
w(i,j,10)>0 OR FN f(ABS s(i,j,10)) THEN GO TO
00350
00315 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30; CHR$ ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0: LET jj=j: GO
SUB 8600: IF LEN v$<>2 THEN GO TO 8350
00320 LET w#=v$: LET par1=j: LET enc=0: FOR c=1
TO 9: LET jj=c: IF jj<>j AND (s(ii,jj,10)<1 AND
NOT FN f(ABS s(ii,jj,10))) THEN GO SUB 8600: IF
w#=v# THEN LET par2=c: LET enc=enc+1
00330 NEXT c: IF enc<>1 THEN GO TO 8350
00332 LET va1=VAL w$(1): LET va2=VAL w$(2): FOR
c=1 TO 9: IF c<>par1 AND c<>par2 THEN LET
s(i,c,va1)=0: LET s(i,c,va2)=0
00335 IF c=par1 OR c=par2 THEN LET
w(i,c,10)=s(i,c,10)-1
00340 NEXT c
00350 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1; " ": IF INKEY$="" THEN LET torol=1: RETURN
00355 NEXT j: NEXT i
00360 IF va1<>0 THEN RETURN
00370 FOR j=1 TO 9: LET ccp=7+(j-1)*2: LET jj=j:
FOR i=1 TO 9: LET cfp=3+(i-1)*2: PRINT AT
cfp,ccp; OVER 1; INK 8; INVERSE 1; " ": IF
w(i,j,10)>0 OR FN c(ABS s(i,j,10)) THEN GO TO
00410
00375 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30; CHR$ ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0: LET ii=i: GO
SUB 8600: IF LEN v$<>2 THEN GO TO 8410
00380 LET w#=v$: LET par1=i: LET enc=0: FOR f=1
TO 9: LET ii=f: IF ii<>i AND (s(ii,jj,10)<1 AND
NOT FN c(ABS s(ii,jj,10))) THEN GO SUB 8600: IF
w#=v# THEN LET par2=f: LET enc=enc+1
00390 NEXT f: IF enc<>1 THEN GO TO 8410
00392 LET va1=VAL w$(1): LET va2=VAL w$(2): FOR
f=1 TO 9: IF f<>par1 AND f<>par2 THEN LET
s(f,j,va1)=0: LET s(f,j,va2)=0
00395 IF f=par1 OR f=par2 THEN LET
s(f,j,10)=s(f,j,10)-2
00400 NEXT f

```

```

8410 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1;" " : IF INKEY$=" " THEN LET torol=1: RETURN
8415 NEXT i: NEXT j
8420 IF va1<>0 THEN RETURN

```

3x3-as blokkok ellenőrzése,és javaslat

```

8422 REM kilences ( 3x3) blokk
8426 FOR m=1 TO 9 STEP 3: FOR n=1 TO 9 STEP 3
8428 FOR i=m TO m+2: LET cfp=3+(i-1)*2: LET
ii=i: FOR j=n TO n+2: LET ccp=7+(j-1)*2: PRINT
AT cfp,ccp; OVER 1; INK 8; INVERSE 1;" "
8430 IF s(i,j,10)>0 OR FN b(ABS s(i,j,10)) THEN
GO TO 8452
8432 BRIGHT 1: FOR k=1 TO 9: PRINT AT 3+k-
1,30;CHR$ ((32 AND s(i,j,k)=0)+(48+k AND
s(i,j,k)<>0)): NEXT k: BRIGHT 0
8434 LET jj=j: GO SUB 8600: IF LEN v$<>2 THEN GO
TO 8452
8436 LET w$=v$: LET par1i=i: LET par1j=j: LET
enc=0
8438 FOR f=m TO m+2: FOR c=n TO n+2: IF f<>par1i
AND c<>par1j AND (s(f,c,10)<1 AND NOT FN c(ABS
s(f,c,10))) THEN LET ii=f: LET jj=c: GO SUB
8600: IF w$=v$ THEN LET par2i=f: LET par2j=c:
LET enc=enc+1
8440 NEXT c: NEXT f
8442 IF enc<>1 THEN GO TO 8452
8444 LET va1=VAL w$(1): LET va2=VAL w$(2): FOR
f=m TO m+2: FOR c=n TO n+2
8446 IF (f<>par1i OR c<>par1j) AND (f<>par2i OR
c<>par2j) THEN LET s(f,c,va1)=0: LET
s(f,c,va2)=0
8448 IF (f=par1i AND c=par1j) OR (f=par2i AND
c=par2j) THEN LET s(f,c,10)=s(f,c,10)-4
8450 NEXT c: NEXT f
8452 PRINT AT cfp,ccp; OVER 1; INK 8; INVERSE
1;" " : IF INKEY$=" " THEN LET torol=1: RETURN
8453 NEXT j: NEXT i
8454 NEXT n: NEXT m: RETURN

```

a választott menüből való visszalépés,mindig az előző menüt tölti be

```

8500 REM választott menü
8510 PRINT #0; BRIGHT 1; INK 0; PAPER 6;AT
0,0;m$;AT 1,0; BRIGHT 0; INVERSE 1;"ENTÉR-re az
előző menüt tölti be"
8520 PAUSE 0: LET t$=INKEY$
8530 FOR t=1 TO LEN o$: IF t$=o$(t) THEN LET
valaszt=t: RETURN
8540 NEXT t
8550 IF t$=CHR$ 13 THEN LET valaszt=0: RETURN
8560 GO TO 8520

```

a hibakeresés után megmutaja a spectrum a lehetséges szám-értékeket

```

0600 REM lista a lehetséges értékekkel v$(ii,jj)
0610 LET v$="": FOR k=1 TO 9: IF s(ii,jj,k)=1
THEN LET v$=v$+STR$ k
0620 NEXT k
0630 RETURN
0700 REM vizszintes kiírás
0710 FOR k=1 TO LEN v$: PRINT AT sor+k-
1,oszlop;v$(k): NEXT k
0720 RETURN
0800 REM a tábla számai 1-5
0810 FOR i=1 TO 5: PRINT AT 2+i,3; PAPER 6+r(i);
BRIGHT 1;i: NEXT i
0820 RETURN
0900 REM meghatározza 1-9-et
0910 LET határoz=0: FOR i=1 TO 9: FOR j=1 TO 9:
IF s(i,j,10)<1 THEN RETURN
0920 NEXT j: NEXT i: LET határoz=1: RETURN

```

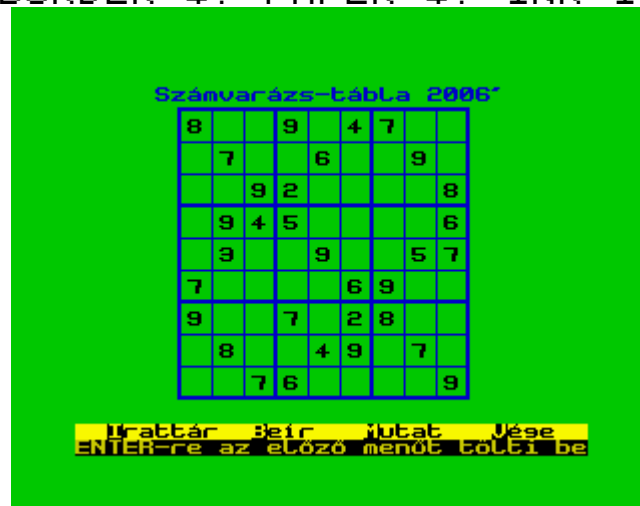
ezzel kezdődik a játék,az elejéről ide ugrik,kiírja a fő-feliratot 'Számvarázs-tábla 2006'



```

9000 REM Számvarázs ( SUDOKU ) felirat
9002 INPUT "": BORDER 4: PAPER 4: INK 1: CLS

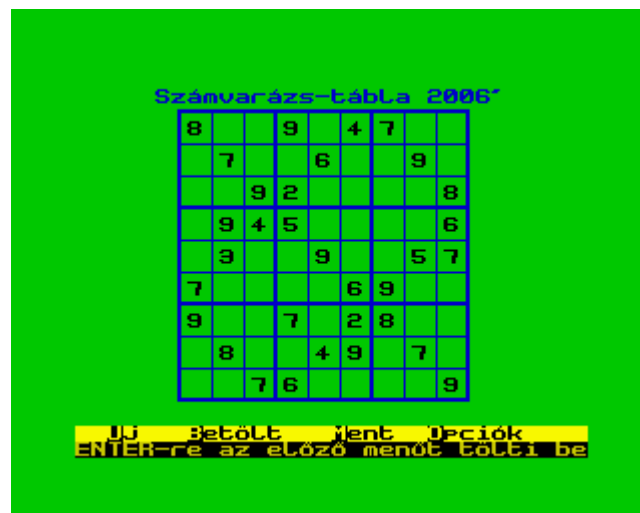
```



a keret zöld,a háttér zöld,a tintaszín kék

```
90003 PRINT INK 1;AT 1,5;"Szamvarazs-tabla
90005": GO TO 9020
90005 PLOT 0,0: DRAW INK 1;255,0: DRAW INK
1;0,175: DRAW INK 1;-255,0: DRAW INK 1;0,-175
90006 LET xi=20: LET yi=170: INK 1
90007 PLOT xi+32,yi: DRAW -32,0: DRAW 0,-4: DRAW
32,0: DRAW 0,-4: DRAW -32,0
90008 LET xi=xi+36: PLOT xi,yi: DRAW 0,-8: DRAW
32,0: DRAW 0,8
90009 LET xi=xi+36: PLOT xi+32,yi: DRAW 0,-8:
DRAW -32,0: DRAW 0,4: DRAW 32,0
90010 LET xi=xi+36: PLOT xi,yi: DRAW 32,0: DRAW
0,-8: DRAW -32,0: DRAW 0,8
90011 LET xi=xi+36: PLOT xi,yi: DRAW 0,-8: DRAW
0,4: DRAW 16,0: DRAW 4,4: DRAW 12,0: PLOT
xi+16,yi-4: DRAW 4,-4: DRAW 12,0
90012 LET xi=xi+36: PLOT xi,yi: DRAW 0,-8: DRAW
32,0: DRAW 0,8
90020 INK 1
90030 RETURN
```

kirajzolja kék színnel a kockás táblát,a 3x3-as tömböknek erősebb a vonala,jól látható módon



```
90070 REM tabla kirajzolas
90075 INK 1
90080 FOR i=0 TO 9
90090 PLOT 52,(3+(i-1)*2)*8+4: DRAW 16*8,0
90095 IF i/3=INT (i/3) THEN PLOT 52,(3+(i-
1)*2)*8+5: DRAW 16*8,0
90100 NEXT i
90110 FOR i=0 TO 9
90120 PLOT (7+i*2)*8-4,12: DRAW 0,16*8+1
90125 IF i/3=INT (i/3) THEN PLOT (7+i*2)*8-5,12:
DRAW 0,16*8+1
90130 NEXT i
90135 INK 0
90140 RETURN

90160 IF alaptabla=0 THEN RETURN
90170 FOR i=1 TO 9
90180 FOR j=1 TO 9
90190 PRINT AT 3+(i-1)*2,7+(j-1)*2;
```

```

00193 IF s(i,j,10)>0 THEN PRINT s(i,j,10): GO TO
00197
00195 PRINT " "
00197 NEXT j
00198 NEXT i
00199 RETURN

```

az előzőleg beírt,vagy betöltött táblát tudjuk megváltoztatni,a fő-maradandó számok feketével,amit mi,vagy a spectrum később ír be -jól elkülönülő - kék színnel

```

00200 REM javitas a tablän
00210 LET cf=1: LET cc=1

```

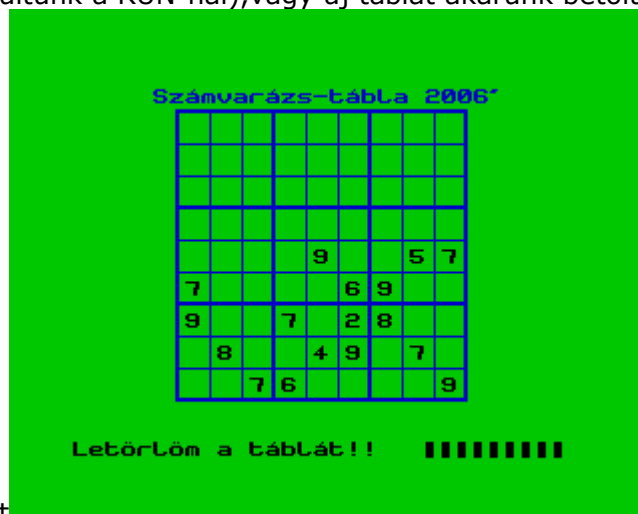
a táblán való mozgás a spectrum legendás irányítási módja :QAOP SPACE,vagyis Q=fel,A=le,O=balra,P=jobbra,0-vagy SPACE tűz(beírás) gomb (az emulátort beállíthatjuk a kurzorgomb irányításra)

```

00225 PRINT #0;AT 0,0; BRIGHT 1;"SPACE torol.
ENTER visszalep."/"VILLOG/QAOP lepes. 1-9
jelöl."
00230 LET cfp=3+(cf-1)*2: LET ccp=7+(cc-1)*2
00235 PRINT AT cfp,ccp; OVER 1; FLASH 1;" "
00240 PAUSE 0: LET t%=INKEY$
00250 IF t%=CHR$ 8 OR t%="o" THEN PRINT AT
cfp,ccp; OVER 1;" ": LET cc=cc-1: IF cc<1 THEN
LET cc=9
00260 IF t%=CHR$ 9 OR t%="p" THEN PRINT AT
cfp,ccp; OVER 1;" ": LET cc=cc+1: IF cc>9 THEN
LET cc=1
00270 IF t%=CHR$ 10 OR t%="a" THEN PRINT AT
cfp,ccp; OVER 1;" ": LET cf=cf+1: IF cf>9 THEN
LET cf=1
00280 IF t%=CHR$ 11 OR t%="q" THEN PRINT AT
cfp,ccp; OVER 1;" ": LET cf=cf-1: IF cf<1 THEN
LET cf=9
00290 IF t%=" " THEN PRINT AT cfp,ccp;" ": LET
s(cf,cc,10)=0: FOR i=1 TO 9: LET s(cf,cc,i)=1:
NEXT i
00300 IF t%=CHR$ 13 THEN PRINT AT cfp,ccp; OVER
1;" ": RETURN
00310 IF t%>="1" AND t%<="9" THEN LET va=CODE t%-
48: LET s(cf,cc,10)=va: PRINT AT cfp,ccp;va: FOR
i=1 TO 9: LET s(cf,cc,i)=(1 AND i=va) OR (0 AND
i<>va): NEXT i
00499 GO TO 0230

```

amikor kezdünk (és nem indultunk a RUN-nal),vagy új táblát akarunk betölteni,akkor



használjuk-e menű-t

```

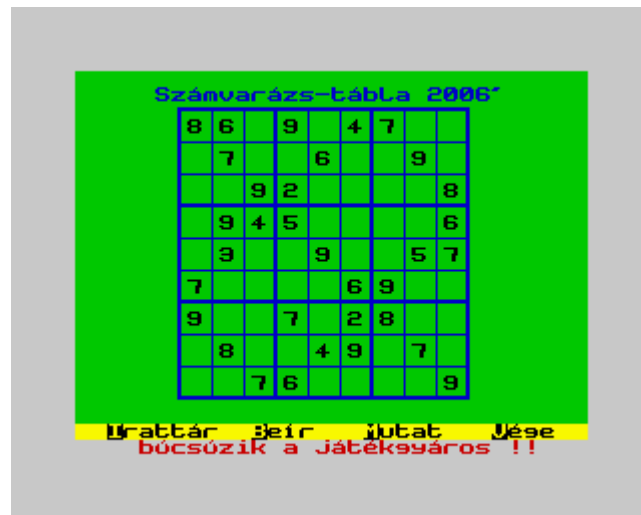
9500 REM toblatorles-memoriatorles

```

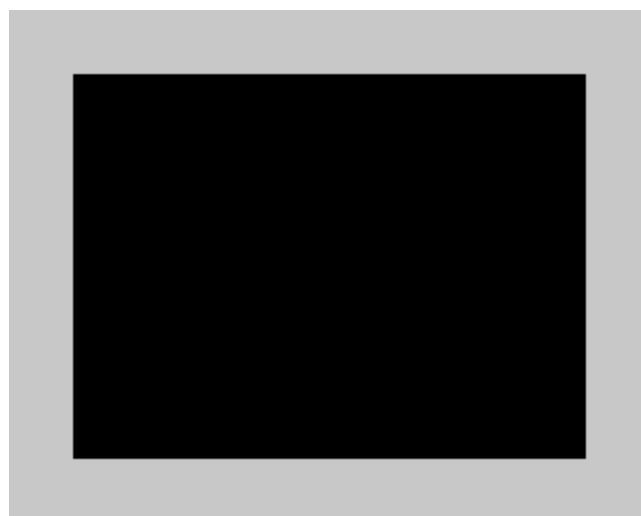
```

9505 INPUT "": PRINT #0;AT 1,0;"Letorlom a
tablat,nullazom a memoriat!!";AT 1,22;
9507 DIM s(9,9,10)
9510 FOR i=1 TO 9
9515 PRINT #0;" ";
9520 FOR j=1 TO 9
9540 FOR k=1 TO 9
9550 LET s(i,j,k)=1
9560 NEXT k
9570 NEXT j
9580 NEXT i
9590 RETURN

```



és akkor elérkeztünk a program utolsó soraihoz,itt kiléphetünk a játékból a játégyáros köszönetével..
'búcsúzik a játégyáros !' felirat a spectrum minden színében,és hangjelzés kíséretében,és utána a borzasztó FEKETE képernyő...



```
9998 REM kilépes a jatekbol
9999 FOR i=1 TO 15: BORDER 7: PAPER 7: INK 0:
BRIGHT 0: FLASH 0: PRINT #0; INK i/2; AT 1,0;"
bucsuzik a jatekgyaros !! " : RANDOMIZE USR
65055: NEXT i: RANDOMIZE USR 0
```

Közreadta :Nyitrai László,akit játékgyarosnak is hívnak...
honlapom <http://jatekgyaros.fw.hu>

Függelék:

Gould könyvei :Egyedülálló számok

A sudoku egyszerűbb elődjét a 18. század végén alkotta meg Leonhard Euler svájci matematikus, mai formáját először 1979-ben publikálták Number Place (számelhelyezés) néven New Yorkban. Japánban 1984-ben adták ki először, de csak 1986-ban lett hirtelen népszerű. 1989-ben Commodore 64-re is megjelent egy sudoku program DigitHunt címmel. A mostani világméretű közkedveltség annak köszönhető, hogy az új-zélandi Wayne Gould, aki korábban bíró volt Hongkongban, 1997-ben Japánban találkozott egy ilyen rejtvényel, és nekiállt kifejleszteni egy szoftvert, ami képes gyorsan előállítani a rejtvényeket. Amikor kész volt, jelentkezett a brit Times-nál, ahol tavaly november 12-én meg is jelent az első sudoku, ami rövid időn belül sikertörténeté vált, ésatöbbi.

A sudoku szó "egyedülálló számok"-at jelent japánul, a kifejezés a Nikoli kiadó birtokában van, legalábbis a rejtvényel kimerítően foglalkozó Wikipedia-szócikk szerint, emiatt az ország többi kiadója inkább eredeti nevén, nanopureként jelenteti meg a fejtörőt (ez a kifejezés "számelhelyezést" jelent).

SZÓREJTŐ – MEMÓRIA - JÁTÉK

NYITRAI-SPECTRUM 128K BASIC-ben.

Ez a program teleírja a képernyőt találomra előhúzott betűkkel, elrejtve benne az előzőleg, a tábla beírt szavakat, melyek jelen esetben 20 darab 5-9 betűhosszúságú, naptárból kiválasztott keresztnevek.

A szövegben elrejtett neveket nehéz megtalálni, mert nyolc irányban, vízszintesen és fordítva, függőlegesen és fordítva, valamint átlósan is négy irányban lehetséges az elhelyezkedésük. A játék menetében mindig lehet a képernyőtartalmat printerre küldeni, a feladatot papírról is meg lehet oldani, mint egy keresztrejtvényt, csak egy szövegekijelölő filctoll szükséges hozzá.

A programozás feladatai, az alábbi szubrutinokat kell elkészíteni:

FŐPROGRAM
ÓRANULLÁZÁS
IDŐMÉRÉS
EREDMÉNYHIRDETÉS
A TÁBLA
JÁTÉK MAGYARÁZATA
JÁTÉKMEZŐ MÉRETÉNEK KIVÁLASZTÁSA
SZÓBEÍRÁS
SZÓRAKTÁROZÁS
A BEÍRT SZAVAK KEVERÉSE
TÁBLA MEGJELENÍTÉS A KÉPERNYŐN
SZÓ-LISTA MEGJELENÍTÉS
SZAVAK - PROGRAM MENTÉSE
MEGJELENÍTÉS NYOMTATÓN

```
10 REM NYITRAI-BASIC
20 REM POKE 23606,150
30 REM POKE 23607,162
40 REM ÉKEZETES BÉTŰK
50 REM Á É I Ó Ő Ű Ú Ü
60 REM BETUKOD 41852,768
70 REM UDG CODE USR "A",168
80 REM *****
90 REM *
100 REM * SZÓREJTŐ JÁTÉK *
110 REM *
120 REM * NYITRAI 1999 *
130 REM *
135 REM * 3 OKTÁVOS HANGGAL *
137 REM *
140 REM *****
150 BORDER 1: PAPER 1: INK 6: C
LS
160 REM ALAPBEÁLLÍTÁSOK:
170 POKE 23658,8: REM NAGYBETŰ
180 CLEAR 64999:
128 BASIC
```

A program Sinclair basic-kel készült, kombinálva a saját nagybetűs basic-el, 128 k-s teljes képernyős szerkesztő móddal, sztereó, háromoktávós hanghatásokkal.

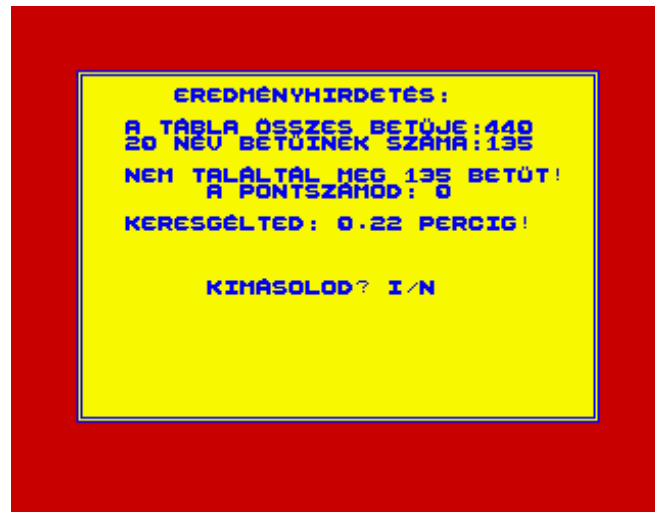
```
1 REM NYITRAI-128K BASIC
2 REM POKE 23606,150
3 REM POKE 23607,162
4 REM ÉKEZETES BÉTŰK
5 REM 01 09 00 13 16 1A Ó
6 REM BETUKOD 41852,768
7 REM UDG CODE USR "A",168
```



```

10 REM *****
20 REM *
*
30 REM * SZOREJTŐ JATEK *
35 REM * 3 OKTAVOS HANG *
40 REM * NYITRAI 1999 *
48 REM *
*
50 REM *****
60 BORDER 1: PAPER 1: INK 6: CLS
100 REM ALAPBEALLITÁSOK:
110 POKE 23658,8: REM NAGYBETŰ
120 CLEAR 64999: REM MEMORIATORLES

```



ha nem kívánunk új szavakat írni, felhasználhatjuk az előzőleg kimentetteket

```

125 REM
499 REM *****
500 REM *
501 REM * FŐPROGRAM *
503 REM *
504 REM *****
510 GOSUB 1000
515 CLS: PRINT AT 1,3; "BETÖLTESZ ADATOKAT?":
PAUSE 0: LET I$=INKEY$: IF I$="I" THEN GOSUB
6200: GOSUB 5860: GOTO 530

```

kellemes hangzású, kis "ping" hang

```
517 PLAY "UX6000W7C"
```

indítás a szavak beírásával

```
520 GOSUB 5000
```

```

300 REM *
310 REM *      ORANULLAZAS
320 REM *
330 REM *****
340 POKE 23672,0: POKE 23673,0:
      POKE 23674,0
350 LET PONT=0
360 LET C1=0
370 LET C2=0
380 PLAY "UX6000W7C"
390 IF E$(C1+1,C2+1)=" " THEN P
PRINT PAPER 2; FLASH 1; AT
C1,C2;A$(C1+1,C2+1)
400 IF E$(C1+1,C2+1) <> " " THE
N PRINT INK 2; FLASH 1; AT
C1,C2;e$(C1+1,C2+1)
410 LET i$= INKEY$
420 IF i$="5" AND C2>0 THEN P
PRINT AT C1,C2;A$(C1+1,C2+1
); INK 3; AT C1,C2;e$(C1+1,
C2+1) AND e$(C1+1,C2+1) <>
128 BASIC

```

minden induláskor a számítógép belső óráját lenullázzuk,a játék megoldásáig,végig méri az időt

```

530 REM *****
531 REM *
532 REM *      ORANULLAZAS      *
534 REM *
535 REM *****
540 POKE 23672,0: POKE 23673,0: POKE 23674,0

```

a szerzett pontszám induláskor nulla,a kurzorpozíció a bal felső sarokban van

```

550 LET PONT=0
560 LET C1=0
570 LET C2=0

```

kellemes hangzású,kis "ping" hang

```
572 PLAY "UX6000W7C"
```



a betűkön való mozgáskor más színnel jelöli a megtalált jó,rossz és semleges szavakat,a szövegben való mozgás a Q-FEL,A-LE,O-BALRA,P-JOBBRA betűkkel,beírás az ENTER-rel

```

582 IF E$(C1+1,C2+1)=" " THEN PRINT PAPER 2;
FLASH 1; AT C1,C2;A$(C1+1,C2+1)

```

```

585 IF E$(C1+1,C2+1) <> " " THEN PRINT INK
2; FLASH 1; AT C1,C2; e$(C1+1,C2+1)
590 LEFT i$=INKEY$
600 IF i$="O" AND c2>0 THEN PRINT AT
C1,C2; a$(c1+1,c2+1); INK 3; AT
c1,c2; e$(C1+1,C2+1) AND e$(c1+1,c2+1) <> " ":
LET c2=c2-1: GOTO 580
610 IF i$="A" AND c2<fugg-1 THEN PRINT AT
C1,C2; a$(c1+1,c2+1); AT c1,c2; INK
3; e$(C1+1,C2+1) AND e$(c1+1,c2+1) <> " ": LET
c1=c1+1: GOTO 580
620 IF i$="Q" AND c1>0 THEN PRINT AT
C1,C2; a$(c1+1,c2+1); AT c1,c2; INK
3; e$(C1+1,C2+1) AND e$(c1+1,c2+1) <> " ": LET
c1=c1-1: GOTO 580
630 IF i$="P" AND c2<vissz-1 THEN PRINT AT
C1,C2; a$(c1+1,c2+1); AT c1,c2; INK
3; e$(C1+1,C2+1) AND e$(c1+1,c2+1) <> " ": LET
c2=c2+1: GOTO 580
640 IF i$=CHR$ 13 AND e$(C1+1,C2+1)=" " THEN
LET e$(c1+1,c2+1)=a$(c1+1,c2+1): GOTO 580
650 IF i$=CHR$ 13 AND e$(C1+1,C2+1) <> " "
THEN LET e$(c1+1,c2+1)=" ": GOTO 580
660 IF i$="U" THEN GOTO 800
670 IF i$="M" THEN GOSUB 9800: GOTO 580
690 GOTO 590

```

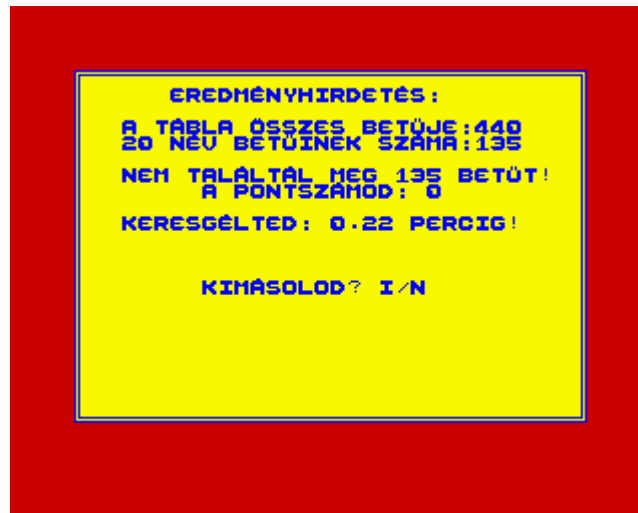


ha végigmentünk az egész táblán,vagy megszakítottuk a játékot,akkor az értékelés következik

```

6005 REM *****
6007 REM * JATEK VEGE *
6008 REM * IDŐMERES *
6009 REM *****
23673*2 6002 LET IDO=(INT ((PEEK 23672+PEEK
C1,C2; INVERSE 1;E$(C1+1,C2+1) THEN PRINT AT
6004 IF E$(C1+1,C2+1)=" " THEN PRINT AT
C1,C2; A$(C1+1,C2+1)
6005 BORDER 2: PAPER 6: INK 2
6006 PRINT #0; AT 0,0; "
..
8030 GOSUB 6000
8000 BEEP .1,25: PRINT AT 21,21;"KIMASOLOD?":
PAUSE 0: LET I$=INKEY$: IF I$="I" THEN GOSUB
9800

```



a megtalált betűk száma, illetve a kihagyottak száma, időmérés, és értékelés

```
890 CLS  
895 REM *****  
896 REM * EREDMÉNYHIRDETÉS *  
897 REM *****  
900 PRINT AT 1,8;"EREDMÉNYHIRDETÉS:";AT  
2,8;"-----"
```

kellemes hangzású, kis "ping" hang

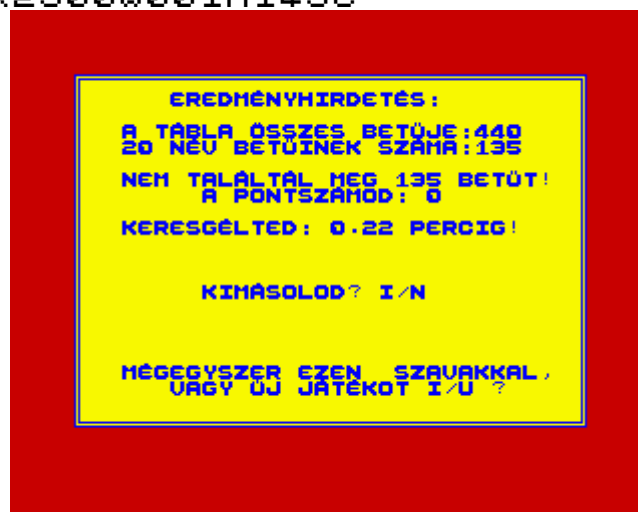
```
901 PLAY "UX6000W7C"
```

Kírja eredményeinket

```
905 PRINT AT 3,1;"A TÁBLA BETŰINEK SZÁMA=  
";vissz#fU99;;AT 4,1;"A 20 NEV BETŰINEK  
SZÁMA:135"  
910 PRINT AT 7,1;"NEM TALÁLTAL MEG ";135-  
pont;" BETŰT";AT 8,2;"PONTSZÁMOD: ";pont;" PONT"  
911 PRINT AT 10,1;"KERESGÉLTED: ";ido;"  
PERCIG"
```

sztereó hangjelzés, egy kis utózóngerővel

```
912 PLAY" UX2500W001M145C"
```

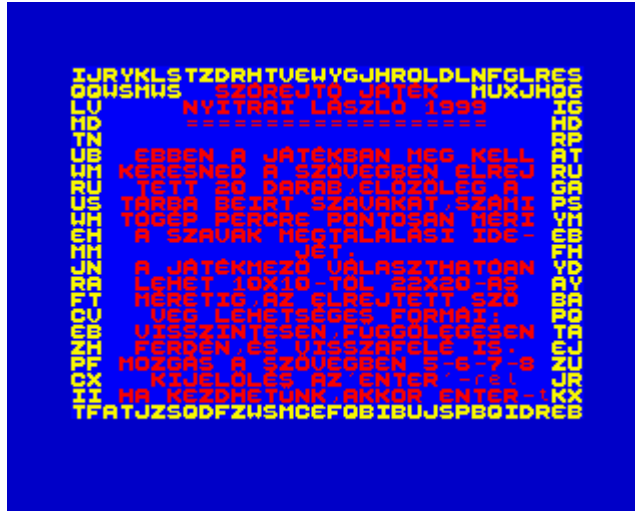


újabb játék kérése,maradjunk a már beírt szavaknál

```
920 PRINT AT 15,5;"KIMASOLOD? I/N"  
930 PAUSE 0: LET I$=INKEY$: IF I$="I" THEN  
GOSUB 9800  
950 PRINT AT 19,1;"MEGEGYSZER EZEKEL A  
SZAVAKKAL?"  
I/N"
```

kis háromtávós hangzón

```
955 PLAY" T240UX50007N2CDECDECDECDECDECDEC"  
960 PAUSE 0: LET I$=INKEY$: IF I$="N" THEN GOTO  
515  
970 IF I$="I" THEN CLS : GOSUB 5330: GOTO 530  
990 STOP
```



a játék valójában itt kezdődik,először betűsorokat ír ki a képernyőre (22 függ.sor * 32 vissz.sor =összesen 704 darab betűt),majd egy ablakot nyit,és erre írja ki a játék magyarázatát,érdekesen, betűként,egymás mellé,hangjelzés közepette.

```
1000 REM *****  
1001 REM * A TABLA *  
1002 REM *****  
1005 BORDER 1: PAPER 1: INK 6: CLS  
1010 FOR N=1 TO 704  
1020 PRINT CHR$( 65+INT (RND*26));  
1030 NEXT N  
1035 BRIGHT 1  
1040 FOR I=2 TO 19  
1045 PRINT PAPER 1; INK 2; AT I,2;"  
..  
1050 NEXT I  
1051 RESTORE 1060  
1052 FOR N=1 TO 18: READ C$  
1053 PRINT AT 1+N, (16-LEN C$/2);  
1054 INK 2: FOR I=1 TO LEN C$  
1055 PRINT C$(I);: BEEP .01,-10  
1056 PAUSE 1: INK 2: NEXT I  
1057 NEXT N  
1060 DATA "SZOREJTŐ JATEK"  
1070 DATA "NYITRAI LASZLO 1999", ""  
1100 DATA "EBBEN A JATEKBAN MEG KELL  
KE", "RESNED A SZOVEGBEN ELREJTETT", "20  
DARAB, ELŐZŐLEG A TARTAB BE", "IRT  
SZAVAKAT, SZAMITGEP PERC", "RE PONTOSAN MERI A  
SZAVAK", "MEGTALALASI IDEJET."  
1105 PRINT
```

```

1110 DATA "A JATEKMEZŐ VALASZTHATÓAN LE", "HET
10X10-TŐL 22X20-AS MIRE", "TIG, AZ ELREJTETT
SZOVEG ", "LEHETSEGES FORMAI: "
1115 DATA
"VISSZINTESEN, FÜGGŐLEGESEN", "FERDEN, ES
VISSZAFELE IS."
1120 DATA "MOZGAS A SZOVEGBEN Q-A-O-P-", "A
KIJELOLES AZ 'E'-REL"
1130 DATA " HA KEZDNETUNK, AKKOR ENTER-t"
1135 BRIGHT 0
1140 PAUSE 0
1150 RETURN

```



a játékmező elkészítése, itt adjuk meg, hogy milyen méretű legyen a szótáblánk, mérete lehet 10x10-től 22x20-es méretű.

```

5000 REM *****
5001 REM #JATEKMEZŐ ELKESZITISE *
5002 REM *****
5005 BORDER 6: PAPER 6
5006 BRIGHT 0: INK 9: CLS
5007 REM *****
5010 REM *10X20 ES 22X20 MERET *
5012 REM *****
5020 CLS
5021 LET h$="FÜGGŐLEGESVISSZINTES"
5030 PRINT AT 1,1;"MILYEN MÉRETŰ LEGYEN A
PALYA?"; AT 2,1;"1=10 ES 20, 2=22 ES 20"
5040 FOR N=1 TO 2
5050 INPUT (H$(N+((N=2)*9) TO
(N=1)*10+(N=2)*20)); " "; LINE X$
5051 IF X$="" THEN GOTO 5050
5060 FOR M=1 TO LEN X$
5070 IF X$(M) <"0" OR X$(M) >"9" THEN GOTO 5050
5080 NEXT M
5090 IF N=1 THEN LET függ=VAL X$: IF függ <10
OR függ >22 THEN GOTO 5050
5100 IF N=2 THEN LET vissz=VAL X$: IF vissz <10
OR vissz >20 THEN GOTO 5050
5110 NEXT n
5120 PRINT : PRINT PAPER 5;" "; függ;" *
"; vissz;" NAGYSÁGU PÁLYÁT"; "
VALASZTOTTAL "
5130 PRINT

```

MILYEN MÉRETŰ LEGYEN A PÁLYA?
1=10 ÉS 20, 2=22 ÉS 20

15 * 15 NAGYSÁGŰ PÁLYÁT
VALASZTOTTAL

MAXIMÁLIS SZÓHOSSZÚSÁG 1-9 LEHET

SZAVAK DARABSZÁMA? 5-20 20

itt írjuk be,hogy milyen szavakat kell majd megtalálnunk,5-és 20 darab lehet,szóhosszúságuk pedig 1-és 9 között lehet.

```
5140 PRINT "MAXIMALIS SZOHOSSZUSAG 1-9 LEHET
";
5150 INPUT "SZAVAK DARABSZAMA? 5-20 "; LINE X$
5151 IF X$="" THEN GOTO 5150
5160 FOR n=1 TO LEN X$
5170 IF X$(n)<"0" OR X$>"9" THEN GOTO 5150
5180 NEXT n
5190 LET darab=VAL X$
5200 IF darab<5 OR darab>20 THEN GOTO 5150
5210 PRINT : PRINT PAPER 5;darab;" SZOT
VALASZTOTTAL"
5210 REM *****
5220 REM * SZOBEIRAS *
5230 REM *****
5240 PRINT
5250 LET w$=""
5260 DIM w(darab,2)
5270 FOR n=1 TO darab
5280 INPUT (n);".SZO: "; LINE X$
5290 IF LEN X$>fugg AND LEN X$>vissz THEN
PRINT #0; FLASH 1;"TUL HOSSZU": PAUSE 300: GOTO
5300
5310 FOR M=1 TO LEN X$
5320 REM IF X$(m)<"A" OR X$(m)>"Z" THEN GOTO
5330
5340 NEXT m
5350 LET w$=w$+X$
5360 LET w(n,2)=LEN w$
5370 LET w(n,1)=LEN w$-LEN X$+1
5380 REM
5390 PRINT PAPER 4;n;". ";w$(w(n,1) TO
w(n,2))
5400 NEXT n
```



a beírt szavak alapján a számítógép összekeveri, elrejtja a szövegben a keresett szavakat, hangjelzés közepette

```

5325 CLS
5328 REM
*****
5330 REM *      A  BEIRT  SZAVAK  KEVERESE
*
5332 REM
*****
5335 PRINT INK 9;AT 10,3;"KEVEREM A SZAVAKAT
!!!"
955 PLAY" T240UX50007N2CDECDECDECDECDECDEC"

```



az előzőleg beírt szavakat a számítógép a DIM tömbökben tárolja

```

5340 DIM a$(fugg,viSSz)
5350 DIM e$(fugg,viSSz)
5355 DIM l$(fugg,viSSz)
5360 FOR n=1 TO darab
5365 PLAY" T240UX50007N2CDECDECDECDECDECDEC"

```

a beírt neveket az RND véletlenszámgenerátorral pozícionálja, az 'r1' és 'r2' a ferde eltérés

```

5370 LET x=1+INT (RND*fugg)
5380 LET y=1+INT (RND*viSSz)
5390 LET r1=-1+INT (RND*3)

```



```

5400 LET r2=-1+INT (RND*3)
5405 IF r1=0 AND r2=0 THEN GOTO 5390
5410 LET xs=(w(n,2)-w(n,1)+1)*r2+x
5420 IF xs<1 OR xs>fugg THEN GOTO 5370
5430 LET ys=(w(n,2)-w(n,1)+1)*r1+y
5440 IF ys<1 OR ys>vissz THEN GOTO 5370
5450 FOR m=w(n,1) TO w(n,2)
5455 BEEP .001,60
5460 IF a$(xs,ys) <> " " AND a$(xs,ys) <>
w$(m) THEN GOTO 5370
5470 LET xs=xs-r2
5480 LET ys=ys-r1
5490 NEXT m

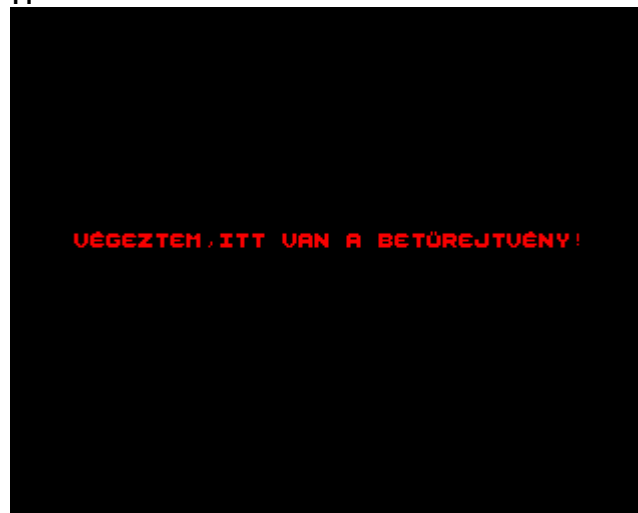
```

a vízszintes után következnek a függőleges eltérítés

```

955 PLAY" T240UX50007N2CDECDECDECDECDECDEC"
5495 LET xs=(w(n,2)-w(n,1)+1)*r2+x
5496 LET ys=(w(n,2)-w(n,1)+1)*r1+y
5550 FOR m=w(n,1) TO w(n,2)
5555 BEEP .001,40
5560 LET a$(xs,ys)=w$(m)
5570 LET l$(xs,ys)=w$(m)
5580 LET xs=xs-r2
5590 LET ys=ys-r1
5600 NEXT m
5610 NEXT n
5620 FOR n=1 TO fugg
5630 FOR m=1 TO vissz
5635 BEEP .001,20
5645 PRINT AT 10,3;"KEVEREM A SZAVAKAT !!!"
5649 IF a$(n,m)=" " THEN LET a$(n,m)=CHR$
(65+INT (RND*26))
5650 NEXT m
5670 NEXT n

```



amikor végzett mindent kiír a képernyőre,először elég nehéznek látszik,mert lehet vízszintesen,függőlegesen,ferdén és visszafelé is az elrejtett szó

```

5678 REM
*****
***
5680 REM * NYOMTATAS-TABLA MEGJELENITES
*
5682 REM
*****
***

```

sztereó orgonazúgás,ha végzett a keveréssel

```
5683 LET X$=" T240UX50007N2CDECDECDECDECDECDEC"
5684 LET Y$=" T240UX50007N4CDECDECDECDECDECDEC"
5685 LET Z$=" T240UX50007N6CDECDECDECDECDECDEC"
5687 PLAY X$,Y$,Z$:PAUSE 100:CLS
```



a keresendő neveket jobboldalt kiírja

```
5690 BRIGHT 1: INK 7: BORDER 0: PAPER 0: CLS
5691 FOR N=1 TO darab
5692 PRINT INK 2;AT 0+N,21;w$(w(n,1) TO
w(n,2)); INK 6;AT 0,21;" NEVEK:"
5693 NEXT N
5700 FOR n=1 TO függ
5710 FOR m=1 TO vissz
5715 PRINT AT n-1,m-1; INVERSE (e$(n,m) <> "
");a$(n,m)
5720 NEXT m
5722 NEXT n
```

lehetőség,hogy papírra is kiirathassuk

```
5727 PRINT #0;AT 0,0;"KIMASOLOD?": PAUSE 0:
LET i$=INKEY$
5730 IF i$="I" THEN GOSUB 9800
5740 PRINT #0;AT 0,0;"ELMENTED AZ ADATOKAT?":
PAUSE 0: LET i$=INKEY$: IF i$="I" THEN GOSUB
6400
5790 PRINT #0; PAPER 5; INK 0;AT
0,0;"MOZGAS:0-FEL,A-LE,0-BAL,P-JOBB,
ENTER=KIJELOL M=MASOL U=VEGE "
5800 RETURN
```



a 'V' betű lenyomásával megszakíthatjuk a játékot, illetve, ha nem találtunk meg minden keresztnévet, akkor a szövegben más színnel megjeleníti az elrejtett szót.

```

5098 REM *****
6000 REM *      ITT A VEGE *
6002 REM *****
6005 LET da=10
6010 FOR n=1 TO fugg
6020 FOR m=1 TO vissz
6025 BEEP .001,10
6026 IF l$(n,m) <> " " THEN LET da=da+1
6030 IF l$(n,m)=e$(n,m) AND l$(n,m) <> " "
THEN LET pont=pont+1
6032 IF l$(n,m) <> " " THEN PRINT AT n-1,m-1;
PAPER (e$(n,m)=" "); INVERSE 1; INK 4;l$(n,m)
6033 IF l$(n,m)=" " AND e$(n,m) <> " " THEN
PRINT AT n-1,m-1; INVERSE 1; PAPER 5; INK 1;
FLASH 1;a$(n,m); LET pont=pont-1
6035 PRINT #0; AT 1,0; "PONT: "; pont
6040 NEXT m
6050 NEXT n
6060 RETURN

```



külön kérésre megjeleníthető a szavak listája

```

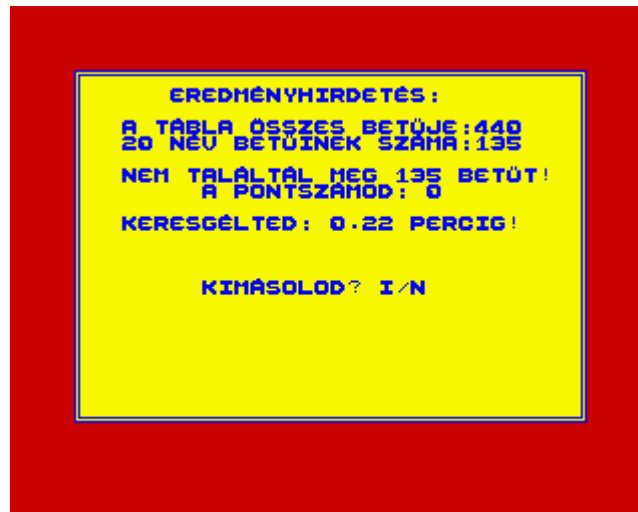
6090 REM *****

```

```

6100 REM *          SZOLISTA          *
6101 REM *****
6110 CLS
6120 FOR N=1 TO darab
6130 PRINT "          " AND n<10;n;".
";w$(w(n,1) TO w(n,2))
6140 NEXT n
6150 PRINT #0;AT 0,0;"KIMASOLOD?"
6160 LET pont=pont-10
6170 LET i$=INKEY$: IF i$="" THEN GOTO 6170
6175 IF i$="I" THEN GOSUB 9800
6180 CLS
6190 RETURN

```



ha előzőleg kimentettük a szavakat,akkor újra ugyanazzal a 20 szóval játszhatunk,a tábla betölthetjük

```

6192 REM *****
6200 REM *          ADATBETÖLTÉS          *
6201 REM *****
6210 LOAD "szo" DATA h()
6220 LET darab=h(1)
6230 LET fuigg=h(2)
6235 LET vissz=h(3)
6240 LOAD "szo" DATA a$()
6250 LOAD "szo" DATA w()
6260 LOAD "szo" DATA l$()
6270 LOAD "szo" DATA h$()
6280 LET w$=h$()
6285 DIM e$(fuigg,vissz)
6290 RETURN

```

itt menthetjük ki a már előzőleg beírt szavakat,nincs nagy jelentősége,hogy más szavakat írjunk be,a régieket is nehéz lesz megtalálni,mert minden új játékban máshova helyezi,újra összekeveri őket

```

6300 REM *****
6400 REM *          ADATMENTÉS          *
6401 REM *****
6410 DIM h(3)
6420 LET h(1)=darab
6421 LET h(2)=fuigg
6422 LET h(3)=vissz
6430 DIM h$(LEN w$)
6440 LET h$()=w$
6450 SAVE "szo" DATA h()
6460 SAVE "szo" DATA a$()

```

```
6470 SAVE "szo" DATA w()
6480 SAVE "szo" DATA l$( )
6490 SAVE "szo" DATA h$( )
6500 RETURN
```

a programlistát is ki kell menteni, indulás a 100-as sortól

```
6600 REM *****
9000 REM * PROGRAM MENTES *
9001 REM *****
9005 CLEAR
9010 SAVE "szo" LINE 100
9030 STOP
```

ha másolás parancsot adunk, ide ugrik végrehajtásra a számítógép

```
9700 REM
*****
9701 REM * MASOLAS NYOMTATORA *
9703 REM
*****
9800 COPY
9801 RETURN
```

Nyitrai László email: jatekgyaros@freemail.hu
Honlap: <http://jatekgyaros.fw.hu>

M Á S O D I K R É S Z

Az informatika iskolai realitásai

Új Pedagógiai Szemle 1998 április - EPA - www.epa.hu Nyomtatóbarát változat:
Országos Közoktatási Intézet > Új Pedagógiai Szemle 1998 április >
Az informatika iskolai realitásai egy középiskolai tanár szemével:
(Nyitrai László r.alezredes,Belügyminisztérium,Informatikai Osztály.)

Az informatika iskolai realitásai egy középiskolai tanár szemével:

A szerző azokra a dilemmákra keres választ, amelyek a számítástechnika iskolai oktatásában hosszabb ideje felmerülnek. Ezen a műveltségterületen ugyanis, a rendkívül gyors technikai fejlődés miatt, nagyon nehéz stabil műveltségtartalmat rögzíteni, tudomásul kell venni a közvetített ismeretek, a fejlesztendő készségek folytonos átalakulását. A tanulmány érinti az Internetnek az oktatásba történő bevonulásával együtt járó sokféle problémát, mindenekelőtt az információk közötti válogatás nehézségeit. Figyelmeztet arra, hogy az informatikai eszközök oktatási alkalmazásakor kísért a veszély, hogy egyedül üdvözítő megoldásként tekintünk az új eszközre.

Nekem mindig az 1982-es év jut eszembe, ha sorsfordulatra gondolok. Ugyanis ekkor találkoztam először azzal a mesterséggel, amit ma számítástechnikának neveznek. Elsőként a mesterség szót kellene megmagyaráznom. Akkoriban nyilván tudománynak neveztem volna, de ma már rá kell jönnöm, hogy egy jó mesterség alkalomadtán többet jelent annál, amit sokan tudománynak neveznek. Ez tehát nem a kiábrándulás kifejezése. Egy mesterember ugyanis tudatában van azoknak a folyamatoknak, amelyek a keze alatt zajlanak. A kőműves tudja, hogy a cement és sóder milyen keverékével kell a ház építésénél a betonkoszorút elkészíteni. Ez tapasztalatból, no meg minden másból számára egyértelmű. Aki az ilyen fogásokat, tevékenységeket elsajátítja, az ismeri a mesterségét.

Nos, mi a helyzet a számítástechnikával?

Azt a titkárnőt, akinek az asztalán ott sutorog a nagy teljesítményű számítógép színes monitorral, és szövegszerkeszt (ma így hívja a gépírást) vagy esetleg táblázatkezelővel dolgozik, nyilván nem nevezhetem tudósnak. Mindenki tudja (ő is), hogy semmi köze nincs a számítástudományhoz. Jó szakmunkásnak is csak akkor mondhatnánk ebben a tekintetben, ha az általa használt szerszám teljesítményének legalább a 10%-át használná. Többnyire még ez sem következik be. Mindössze arról van szó, hogy megtanították a Windows használatára, begyakorolt néhány kattintási sorozatot, amelyből a gyakorlata során mintegy a felét használja is. Ha ugyanannak a szövegszerkesztőnek a fejlettebb változatát kellene használnia, akár el is mehetne a következő tanfolyamra. Ott aztán megtanulja, hogy a fejlettebb változaton hova kell kattintani, hogy a gép az eddig megszokott néhány műveletet végrehajtsa, mert gondoskodtak arról, hogy a fejlettebb változaton ne ugyanott találja meg a megszokott menüpontot. Természetesen nem vitatom, hogy attól, ha valaki a számítógépet csak így kezeli, még lehet jó titkárnő, mint ahogy a kőműves sincs tisztában a cement kötések lezajló kémiai folyamatokkal.

Megpróbálom az előbbieken kissé sarkított jelenséget más szemmel nézni. Amikor 1982-ben először találkoztam a számítástechnikával, akkor az még a matematikához képest csak egyszerű mesterségnek tűnt számomra. Később rájöttem, komoly elméleti munkák, matematikai tételek és bizonyítások állnak mögötte. Igazi tudományággá nőtte ki magát. Ki gondolta volna, hogy az akkori teremnyi méretű óriási kapacitásúnak hitt gépeket ma már bizonyos tekintetben a zsebkalkulátorok is túlszárnyalják? Ugyanakkor a programozás szinte

automatikussá vált. Ma már nem is kell tudnia programozni annak, aki a számítógépet használni akarja. A gépek teljesítménye, kapacitása óriásit nőtt, és egyre fejlettebbek a programok.

A számítógép bevonult a mindennapi életbe. A munkahelyeken, az otthonokban használják munkaeszközként, szórakozásként. Könnyedén, egyszerűen hoznak létre vele távoli kapcsolatokat. Az információ áramlása soha nem látott méreteket ölt. És mindez az embert szolgálja. A fejlődés, vele együtt a tanulás persze nem áll meg, egyre újabb és újabb feladat megoldására használhatjuk a gépet, ami azt jelenti, hogy kezeléséhez újabb és újabb dolgokat kell megtanulni. Mégis van néhány megoldatlan probléma a számítástechnika tanítása körül!

A számítástechnikát tanítók dilemmái

Annak, aki a számítógépes ismereteket tanítja, s azt hivatásának tekinti, a többi tantárgy tanárához hasonlóan azzal a problémával kell szembenéznie, mit tanítson és azt hogyan tanítsa. A mit és hogyan tanítsunk? problémája tekintetében az informatika vagy még inkább ennek a számítástechnika része, igencsak kilóg a tantárgyak sorából. Amíg egy matematika-tanterv esetében a kialakított témakörökön lehet ugyan vitatkozni, lehet súlyozni, például időtartamban, mélységben, a sorrend meghatározásában stb., addig a számítástechnika esetében szembe kell nézni egy más természetű problémával is. Egy jó matematika-tanterv sok évig jó, azért, mert a megtanulandó anyag nem változik az általános vagy a középiskola szintjén. A matematika mint tudomány ma ritkán foglalkozik a középiskolás tanulók szintjére lehozandó problémákkal. Ha mégis, akkor az legfeljebb kis korrekciót jelent a tananyagban. Ennek ellenére egy-egy témakör bevezetése így is kisebb viharokat kavart. A fizika tantárgy már kevésbé van ilyen kellemes helyzetben, de nem sokkal nehezebbek a problémái. Inkább az elmélet és a gyakorlat aránya, a számonkérés módja jelent problémát. Az átlagos diák a Newton-tételek szintjén marad a mechanikában, legfeljebb Faraday kísérleteinek elemzéséig jut el az elektrodinamikában. A kvantummechanika világa csak megcsillan, de a homályos magyarázatok világában marad, mert a diáknak nincs hozzá kézzelfogható, mindennapi tapasztalata. Ennek ellenére a tanterv megváltoztatása nem jelent nagymértékű tananyagváltást. A számítástechnikában azonban eddig mindig az volt a helyzet, hogy két egymás utáni évfolyamon is váltani kellett a tananyag struktúrájában. Körülbelül négyévente ki kellett cserélni a tanítandó technikát is. Így a diák a középiskolai tanulmányai során legalább két számítógép-generációval találkozik. A tananyag ilyen gyors változásának szükségszerűségét természetesen azok is látják, akik a NAT bevezetésével kapcsolatos tevékenységeket végzik. Az informatika-anyagban NAT-szinten nem szerepel a CD-ROM fogalma, és az internet vagy multimédia szó sem domináns, a hálózat kifejezés pedig úgyszólván olyan félve kerül be, mint a matematikába a differenciál- és integrálszámítás. Ugyanakkor a vizsgakoncepcióban már multimédia és CD-ROM megismertetése szerepel alapfeladatként. Ráadásul a tanítványok egy kis része, messze meghaladva tanítóját, a csillagos eget sem tartja határnak, mások pedig alig jutnak túl az egérkattintás művészetén. Elképzelhetjük, hogy ha a NAT-ot tizenöt évvel ezelőtt publikálják, benne a számítástechnika tantárggyal, akkor a kettős számrendszerben történő összeadás vagy a regiszterek közötti adatcsere alapfogalom lett volna, mert akkor sokan azt gondolták, hogy az képviseli a számítástechnika lényegét. Ma, nagyon helyesen, alig említjük őket. A különböző tantárgyak és tantárgycsoportok képviselői pedig, bevallottan, nem tudnak mit kezdeni azzal a kérdéssel, hogy tantárgyukra milyen hatással lesz a számítástechnika.

A fenti problémák világossá teszik, hogy nem lehetséges a számítástechnika tantárgyban időtálló témaköri koncepciót kialakítani. Egyes témaköröket fontosnak tarthatunk az adott pillanatban, másokat mellőzhetünk, majd rövid idő múlva, addig csak érinteni kívánt témakörök fontossá válhatnak, felborítva a kialakult rendet. S akkor még nem is foglalkoztunk az egyes témakörökkel kapcsolatos módszertani problémákkal. Jelenleg a számítástechnikát tanító kolléga kénytelen egyszerre magáravállalni a tantervíró, a megértő pedagógus és a rendszermérnök feladatát. Aki a NAT-hoz kapcsolódó helyitanterv-írásra adja a

fejét, annak előre ki kell jelentenie, hogy legfeljebb a következő tanév az, amíg tanterve becsülettel követhető.

Mindez egyben azt is jelenti, hogy a számítástechnikát tanító tanárnak minden tantervváltáskor majdnem egy új tantárgyat kell megtanulnia. Fejest ugrani kényszerül a hálózatok hálózatába úgy, hogy az alaphálózati ismeretei is hiányosak. Az Internetet pedig vajmi kevésbé ismerheti, legalábbis nem a tanítható szinten. Az Internettel olyan eszköz kerül az iskolába, amelynek természetét a tanárok nem ismerik, használatában pedig legfeljebb egyenrangúak a tanítványainkkal, semmiképp nem gyakorlottabbak náluk. Ez veszélyes vállalkozás, a kimenetele nem minden esetben pozitív. Ez a játszma nem azonos feltételek között zajlik a valamikori iskolaszámítógép-programmal.

Az Internet iskolai hozzáférhetősége felvet pedagógiai természetű gondokat. Bizonyos tények helyes felfogása ugyanis életkorhoz kötött. A velük való idő előtti találkozás veszélyes lehet, ha nem párosul alkalmas, felnőtt módon gondolkodó partner támogatásával. Hiszen az Internet gátlás nélkül mindent elérhetővé tesz, nem törődve azzal, hogy ki ül a másik oldalon a számítógép előtt. Az Interneten megjelenő pornó pedig csak a jéghegy csúcsa vagy ennek a kérdéskörnek látható jele.

Ha bárki arra gondol, hogy az Internetet is be lehet lopni ugyanúgy az iskolába, mint annak idején a számítógépet azzal, hogy letesszük az asztalra, az veszélyes játékba kezd. Az iskolaszámítógép-program ugyanis gépet adott, de szoftvert, vetítésre kerülő lemezt, programot nem, vagy későn és keveset. A kereskedelemnek, reklámaparnak (nem is volt akkoriban nálunk) és minden más egyéb, gazdaságot, morált stb. befolyásoló tényezőnek nem lehetett beleszólnia abba, mi folyik az iskolában a számítógépekkel. Az Internet azonban teljesen kívülről jövő információáradat. A tanárnak nincs módja befolyásolni, hogy mi van rajta, vagyis teljes egészében csak szoftver. Pusztán szerencse kérdése, hogy mindez milyen fejleményekkel jár.

Semmilyen anyagi erőforrás sem sok tehát, amit arra fordítunk, hogy az oktatásban dolgozók megfelelő módon megismerjék ezt a médiát. Nem elegendő az, hogy technikai szempontból kezelni tudják, hanem pontosan kell ismerniük, szakszerűen használniuk, hogy legalább partnerei legyenek tanítványainknak. A technika és használatának elsajátíttatása a diákokkal csak egyik vonatkozása a dolognak. Lehet, hogy talán csak ez a számítástechnika-tanár feladata. Valójában azonban ez a legkisebb gond, legalábbis akkor, amikor már működik a dolog, amikor a feldolgozásra kerülő téma a képernyőn megjelenik, akkor annak a pedagógiai értelmezése inkább a földrajzot, természettudományt, művészeteket tanító kollégára meg az osztályfőnökre tartozna.

Mielőtt azon töprengenénk, hogy a számítástechnika-informatika milyen szerepet tölt vagy tölthet be az iskolában, foglalkozzunk azzal, hogy az iskolarendszerekből már kikerült emberek hogyan állnak az ott megszerzett ismereteikkel.

Néhány gondolat a képzettségről és a műveltségről

Azért tartom fontosnak feszegetni a címbeli kérdéseket, mert a NAT bevezetése és a tudományos-technikai forradalom következményei megfelelő alkalmat és indokot jelentenek arra, hogy végiggondoljuk azokat a megváltozott körülményeket, amelyek hatással lesznek a tanított nemzedék épzettségére, műveltségére, tudására. Számtalanszor vagyunk tanúi annak, hogy az írott és az elektronikus sajtó, a médiák összessége a világ legkülönbözőbb tájainak hírértékű eseményeit úgy közvetítik, hogy azokat saját beállítottságuknak megfelelően megpróbálják a józan ész, saját ismereteik, esetenként a tudomány igazságaival vagy mindezek híján a misztika segítségével megmagyarázni. Értelmezéseik gyakran nem haladják meg a többi emberét. Ha az ilyen átlagosan képzett embertől például egy üstökös várható megjelenése idején megkérdezzük, hogy mi is az üstökös: hogyan kering, mikor látszik és miért, akkor bizonyos kérdésekre jól válaszolnak, másokkal kapcsolatosan abszolút téves az elképzelésük. Vagyis egyes dolgokra emlékeznek az iskolában tanultak alapján, de lényeges összefüggések is hiányozhatnak az ismerettárukból. Ugyanígy állunk az összes lehetséges tudományággal és nem csak a természettudományokkal. Valamennyien laikusok

vagyunk, mert mindegyiknek a mélyére nem láthatunk. Az ismereteink sok területen nagyon felszínesek, számos közülük pedig már régen elavult. Különösen veszélyesnek látom azt a folyamatot, amely a természettudományok visszaszorításában nyilvánul meg. Ez ad ugyanis leginkább teret annak, hogy mindenféle téveszmék lábra kaphassanak. A természettudományos ismeretek visszaszorítása nem egyszerűen azt jelenti, hogy nem tanítjuk a szükséges biológiai, fizikai ismereteket, ennél sokkal bonyolultabb a helyzet. Mivel a technikai praktikizmus igen fontos ahhoz, hogy több témában sikeresek legyünk, ezért gyakran azt tanítjuk, hogy mit kell csinálni ahhoz, hogy valami működjön. Néhány példával szeretném megvilágítani, hogy mire gondolok.

A villanyszerelő szakmunkás rajz alapján meg tudja csinálni egy épület elektromos hálózatát. Kisebb épület esetében némi gyakorlattal talán rajz nélkül is. Nem kell értenie ehhez az elektromosság lényegét. Talán csak annyi ismerettel kell rendelkeznie, hogy ha a vezeték egyik vége feszültséget kap, akkor ugyanaz lesz a másik végén is. Ha tervrajza van, akkor ez az ismeret is hiányozhat, hiszen csak követnie kell a tervet. Ha meggondoljuk, a tervezőnek is sokkal inkább kell igazodnia bizonyos konvenciókhoz, egyezményes jelekhez, szabványos alkatrészekhez, mintsem szakértőnek lennie az elektromosság lényegi tulajdonságai tekintetében. Könnyedén lehet akár számítógépet is összerakni az elektromos áramkörök viselkedésének ismerete nélkül.

Életünk tele van praktikus ismeretekkel. Csak az sikeres, aki ezeket jól használja. Minél bonyolultabb az a kor, amelyben élünk, annál több és összetettebb praktikus ismeretekre kell szert tennünk.

Mit láthatunk mindebből? Bár a praktikus ismeretek fontosak, gyakran nem sokat jelentenek az igazi tudás, műveltség szempontjából. Aki nem látja világosan azokat az összefüggéseket, amelyek a praktikum mögött vannak, az könnyen csúszik át a hibás magyarázatok, a tudomány helyett a misztikum világába. Ezt csak akkor előzhetjük meg, ha az összefüggéseket is látjuk. Az összefüggések megkeresése, felfedezése azonban nem egyszerű feladat. Még nehezebb, hogy megtanítsunk valakit az összefüggések meglátására. Több okból is nehezebb, ráadásul veszélyes. Ugyanis mindig csak a meghatározott jelenségek, folyamatok, események közötti konkrét összefüggés tanítható. Az elvont fogalmak közötti kapcsolatok csak akkor mutathatók fel, ha konkrétumokkal megalapoztunk valamit. A konkrét események azonban a tanár által válogatottak, és ezekkel irányíthatja, hogy milyen összefüggéseket lásson meg a tanítvány.

A témakörhöz tartozó jelenségeket sorolhatnánk még hosszan, például kitérhetnénk arra, hogy értelmetlenné válhat a pusztán praktikus ismeret, amikor új helyzet jön létre. Inkább arra kellene választ keresni, mit lehet tenni annak érdekében, hogy egyre több olyan fiatal ember hagyja el az iskolát, akiket nem lehet megtéveszteni áltudományos fecsegéssel. Olyanok, akik a saját tapasztalatuk alapján tudják, merre akarnak tartani. Mivel segítheti ezt elő a mai iskola? Az emberek közötti kommunikáció az emberiség története során mindig nagyon fontos volt. Mi sem bizonyítja ezt jobban, mint az, hogy mekkorát fejlődött. A taglejtésektől kezdve a barlangrajzokon keresztül az Internetig. Ebben a fejlődésben az a pillanat, amelyet Gutenberg neve fémjelez, volt a legfelemelőbb. Akkor még csak arra lehetett gondolni, hogy az írásbeliség nagyot lépett: hozzásegítette a tömegeket a könyvolvasáshoz. Azóta a nyomdatechnika a színes képeivel, szövegeivel elárasztja a világot. Ugyanakkor korunkban egyre nagyobb információhalmazt kell továbbjuttatni, mégpedig úgy, hogy eljusson az emberi értelemig. Ehhez nem elegendő az írott szöveg. Gyorsabb, hatékonyabb eszközökre van szükség, és azok meg is születtek. A hatékonyabb eszközzel viszont hatékonyabban lehet félrevezetni, ártani is. Amíg a nyomtatott szöveget, az írott sajtó termékeit tetten érhető, felelős személyek írják, addig az internet mint média mindenki által fejlesztett és szerkesztett. Mielőtt elbűvölne bennünket annak a hatalmas demokratizmusnak a lehetősége, amit ez jelent, nézzünk körül, hogy mivé lett az, mióta kereskedelmi médiává vált, amióta működnek az Internet-játékok szerverei.

Abban a korban élünk, amelyet a tudományos-technikai forradalom korának neveznek. Mellette vagy annak részeként lezajlik az információs forradalom is. Mint ahogyan meg kell tanulnunk használni a technikai forradalom által nyújtott eszközöket, ugyanúgy meg kell tanulnunk az információs forradalom áldását is

kezelni. E vonatkozásban azonban korántsem a számítógép kezeléséről kell beszélni. Az másodrendű, harmadrendű kérdés. Az információ kezelésének technikája is megtanulandó, az sem elsődrendű, hiszen merő prakticismus, ami igen fontos, de nem a lényeg. A jelentős az, hogy az információtömeg, az eszközök, amelyeket használunk, azt szolgálják, hogy felismerjük a környezetünkben a lényegét és tegyék teljessé az életünket!

A tanár számára az igazi dilemmát mindig az jelenti, hogy a nagy mennyiségű tudáshalmaz miképpen szelektálható. Ha ugyanis egyfelől túl sok mindenről jutnak el felszínes ismeretek hozzánk, akkor elmaradhatnak azok az elvek, amelyek érthetővé teszik a kapcsolatokat az egyes jelenségek között. Ha pedig másfelől csak elveket tanítunk, akkor nem alapozza meg azokat elegendő és széles látókört nyújtó tényanyag. Úgy látom, hogy az Internet önmagában sem az előbbi, sem az utóbbi alternatívát nem támogatja, de sajnos nem hozza létre a kettő közötti egyensúly állapotát sem. Valami újat jelent. Bizonyos tekintetben közel áll a televízió szerepéhez, de azon is túllépett, amikor a virtuális teret létrehozta. Ez a következő fontos kérdést veti fel:

Az Interneten megjelenő információnak mennyi köze van a valósághoz?

Az Interneten megjelenik mindenféle szöveg, kép, hang, mozgókép stb. Ha felnőttként, jelentős tapasztalattal a való világról nézzük ezeket, akkor elfogadható, hogy ránk bízva a döntést, hogy mit tekintünk csupán a képzelet szüleményének. De ha arra gondolunk, hogy ugyanezt gyerek használja, akkor más a helyzet. Jobban is sarkíthatjuk a dolgot. Így például nem valószínű, hogy a Transformers típusú rajzfilmek, amelyekben félpercenként történik a fizika törvényeinek ellentmondó jelenség, jót tesznek a fizika tudományos gondolkodásmódja kialakulásának. Hasonlóképpen nem biztos, hogy a különböző reklámokban alkalmazott trükkök a valóság képeit sugallják a gyerekeknek. Amikor a valóságot leíró modellt próbáljuk tanítani a diákoknak, az addigi élményeikre kell alapoznunk. A virtuális valóság tere hasonlít a valódihoz, amíg játéknak tekintjük, de egyre több olyan részlete van, ami a képzelet szüleménye. Ennek a hatását nehéz lemérni. A mostani felnőtt generáció gyerekkorában legfeljebb moziba járt, ahol többnyire játékfilmet láthatott, valódi szereplők valódi képével. Az első igazi trükköt alkalmazó játékfilm a bagdadi tolvaj volt, amelyben megjelent az óriás. Senki nem gondolta, hogy az ilyesminek a pillanatnyi élményen kívül más hatása lehet. Most pedig a digitalizált kép, hang stb. tetszőlegesen manipulálható. Ha pedig megvan a manipuláció lehetősége, akkor használják is, hogy minél meghökkentőbb, színesebb, figyelemfelkeltőbb legyen a produkció. Egyre nagyobb mennyiségben zúdul ránk mindez, nem csoda, hogy a tapasztalatlan gyerekek sok trükköt képzelnek valóságnak.

Spielberg filmjeiben megjelenő fantáziavilágra gondolhatnánk. Természetesen nem baj, hogy bizonyos dolgok a mesének látszó világból a valóságban is előkerülnek, ugyanúgy, mint a Verne Gyula által megírt regények esetében. Inkább az, hogy a valóság meghamisítása nem minden esetben párosul jó szándékú fantáziával. A tévében vagy az Interneten megjelenő nyilvánvaló trükkök mennyisége olyan mértékűvé válik, hogy a tapasztalatlanok összekeverhetik a valóságot a számárságokkal.

Az iskola csak úgy veheti fel a harcot a téveszmék, az alaptalan misztikum terjedése ellen, ha a világunkat jól reprezentáló tényeket ismertetünk meg, lehetőleg többféle szemszögből, és ezzel együtt megtanítjuk a gondolkodás és helyes következtetés módját.

A számítógép az oktatás technikai eszközrendszerében.

Minden oktatási folyamat valamiféle információközlés. Lehet, hogy az információ csak annyit jelent, hogy ráirányítja valamire a figyelmet, lehet, hogy memoriterként bebiflálandó szöveg. Sok esetben a metakommunikáció elemei is felhasználásra kerülnek a folyamat során. Sajnos ennek jelentősége kezd háttérbe szorulni, mert egyre több technikai eszközt használunk a tanításban. Nem lehet megkerülni azt a kérdéskört, hogy az informatika milyen oktatási eszköz. A

számítástechnikában és könyvtár-informatikában megtanult ismeretek hogyan alkalmazhatók az egyes tantárgyak oktatásában?

Legjobb lenne az egyes tantárgyak helyett a tanulási folyamatról beszélni. A mai iskolai realitás azonban mégis a tantárgyakat jelenti. A pedagógusok esküsznek ugyan a tanulás-tanítás egységes folyamatára, főként nem a saját tantárgyukkal kapcsolatos esetekben. (A saját tárgyak ugyanis kivételesen fontos és meghatározó. Ez az enyhe szakmai sovinizmus természetes, szakmai ártalom.) Sajnos vagy szerencsére minden tantárgynak megvan a sajátos belső logikája. Ez a lételeme. A tanár szemszögéből tekintve pedig van sajátos tantárgy-pedagógiája. Nem vitathatjuk, hogy a testnevelés, a rajz, az ének vagy a matematika ma különálló tantárgyak. Nem lehet a testnevelést ugyanazokkal a módszerekkel tanítani, mint a matematikát.

Ha egy pedagógus a technikai eszközöket nem egészen szokványos módon alkalmazza, vagy egyszerűen csak használni akarja az adott kor csúcstechnikáját, hirtelen sok munkája lesz. Az első eufórikus hangulatot az írásvetítő bevezetése okozta. Egyetemista koromban a bemutató órán erénynek számított néhány fólia használata. Azt demonstrálta, hogy a kolléga készült az órára, nem csak úgy hevenyészve rajzolta fel a táblára a kísérlet ábráját vagy írta fel a mintapéldát.

Elidőztünk kis ideig a ló túlsó oldalán, mire visszakerültünk az innensőre, megtapasztalva, hogy ismét nem a lovon ülünk, vagy ha igen, akkor nem ilyen lovat akartunk. Mára az írásvetítő az öt megillető helyen található az eszközök között, de nem kötelező tábla és kréta helyett is használni.

A diavetítőt is felfedezhettük rövid időre. A következő nagy eszköz pedig a video volt. Az elsőként megjelenő oktatófilmek sajnos nem igazán illettek a megtervezett oktatási folyamatba. Talán természetüknél fogva, hiszen egy film megszerkesztésének szabályai vannak. A képernyő mégis csak a hisz egy kis szeletét ábrázolja, míg az ember szeme 180 foknál is többet lát be. Legalábbis ilyen módon szűkíti le a látásmódot a video az élő előadáshoz képest. Ma már profi videofilmek állnak rendelkezésre, jól előkészített fóliaszorozatok készíthetők vagy vásárolhatók. Ezek alkotói azonban pontosan tudják és figyelembe veszik, hogy a tanulók ezeket a való világgal kell hogy összevegyék. A tanárnak pedig visszatérő feladata, hogy a látott részletekben az egészre hívja fel a figyelmet. Természetesen fontos a tanár szerepe egy video-oktatófilm elkészítésénél, de nem kell mindenkinek ilyesmit alkotnia. Senki nem veheti komolyan, hogy a jövő heti órájára készülve forgatókönyvet kelljen írnia. Legfeljebb megkeresi azt a filmet, amit egy alkalommal már látott a tévében, ha gondolja, hogy az óráját az majd segíteni fogja. Még így is nagy apparátust kell megmozgatnia az iskolában a vetítés előkészítésére, így nem éri meg mindenféle jelentéktelen esetben. Ez azt jelenti, hogy az óráinak csak kis százalékán fog vetíteni. (Erősen befolyásolja ténykedését, hogy mi módon fér hozzá az eszközökhöz, valamint, hogy mennyire ért azokhoz.)

Mi a helyzet a számítógéppel?

Kissé kerülő úton jutott el az oktatási eszközök sorába. Alkalmazása először mint az oktatás célja szerepelt. Gondolom, még ma is ennek nagyobb a súlya, s talán még elég sokáig így lesz. Csak abban az értelemben eszköz, hogy nehéz lenne nélküle számítástechnikát tanítani.

Azonban az első pillanattól kezdve felmerült a lehetőség, hogy különböző tárgyak tanításához használjuk.

Sok kollégám, köztük magam is készítettem a különféle programokat éjszakákon keresztül, hogy aztán másnap vagy egy hét múlva bemutathassuk tanítványainknak vagy kollégáinknak az eredményt. Készültek matematikai egyenletmegoldó programok, fizikai szimulációk BASIC nyelven. Nyelvoktató programokkal is próbálkoztunk. Sok esetben nem is a tanár kódolta a megoldást adó programot, hanem a diákok. Közülük sokan már egyetemet végzett megbecsült szakemberek. Akkor ez volt az igazi tanulási folyamat azok számára, akik ezt akarták tanulni. Azt hiszem, sokkal kedvezőbb hatású volt a dolog a programok írói számára, mint azokéra, akik ezeket használták.

Mint ahogy minden új lehetőség megjelenésekor úgy véljük, hogy megtaláltuk a bölcsek követ, úgy jelent és jelenik meg ma is a számítógép az oktatási eszközök

között. A mai gépek hatékonyabbak, kibővült a paletta a multimédia lehetőségeivel, és most már itt az Internet is. Permanens módon tartja magát, hol itt, hol ott felbukkanva ugyanaz a jelenség, ami annak idején az írásvetítől kapcsán megjelent. Vagyis az a meggyőződés, hogy megtaláltuk a legjobb oktatási eszközt.

Valóban?

Amit a számítógéppel produkálni lehet, az valóban fantasztikus. A kép, a hang, a sok-sok belső információ együtt jelenik meg. A diák a saját üteme szerint haladhat végig egy bizonyos anyagon, ha akar. A CD-ROM viszonylag olcsó lehetőséget ad rengeteg információ tárolására. Az Internet pedig távoli vidékeken élő emberekkel teremti meg a kapcsolatot. Lehetőséget nyújt sok olyanra, ami eddig nem volt. Nem multimédia csupán. Egy óriási könyvtár, színes újságosstand, minden, mi szem-szájnak ingere. Nem csak információk tárháza a számítógép, mint ahogy a tévé sem tükör csupán, amiben a világ tükröződik. De ugyanúgy, ahogyan eddig is, most is leselkednek a veszélyek. A legfőbb veszélynek az tűnik, hogy ismét átesünk annak a bizonyos lónak a másik oldalára. Ezúttal azonban nem csupán arról van szó, hogy akkor is írásvetítőt fogunk használni, amikor krétát kellene, vagy számítógépet a tanári magyarázat helyett. A veszély máshonnan is leselkedhet. Az Internet ugyanis, mint már említettem, amely az iskolában vagy otthon megjelenik, ugyanúgy nem befolyásolható, mint a tévé. A televízióból azonban könnyen kiemelhetünk olyan filmrészleteket, amiket a tanórán videón bemutatathatunk. Az Internetet használva alig irányítható általunk egy tanulócsoport.

Lehet, hogy a leghatásosabb tanulási eszközzé, módszerré válik. A multimédiától kölcsönzött tulajdonságai miatt az információs folyamatban sok esetben nemcsak fogyasztóként, hanem tevékeny résztvevőként lehetünk jelen. Ugyanúgy, ahogyan nem tervezhető előre, hogy egy Internetes játéknak mi a kimenetele, az sem garantálható, hogy egy tanulócsoport minden tagja az általunk tervezett helyeket járja-e be szörfözés közben. Az ott levő információ pedig a tényeken kívül eszméket, téveszméket, életszemléletet stb. közvetít. Olyanokat is, amelyekkel nem azonosul a mai iskola. Könnyen előfordul, hogy az oda nem illő részletekkel a tanár szeme nem is találkozik, mert olyan sok minden történik, hogy nem veheti észre. A tanuló pedig azzal az érzéssel fogja fel az információt, hogy az a világ általa is követendő részéhez tartozik.

Természetesen mindez nem azt jelenti, hogy az Internetet és az általa közvetítetteket ki kell zárni az iskolából, hanem fel kell készülni arra, hogy jól használjuk. Ha ebből a szempontból nézzük, a kör bezárul. Oktatási eszközeink a táblától a számítógépig: mind szerszámok. Hatásfokuk akkor optimális, ha a legjobb pillanatban alkalmazzuk, de a fő eszköz továbbra is az emberi nyelv, a szituációhoz illeszkedő metakommunikáció, a tanári egyéniség!

MIDWAY-i CSATA - hadműveleti torpedójáték

Ez a játék más mint, az előzőek. A régi típusú torpedójáték, egy 10*10-es táblán játszódik, ez pedig egy 20*20-as, hatalmas, teljes képernyőt betöltő táblával rendelkezik, és a csata nem csak a tengeren, hanem a szárazföldön is folyik, és felváltva látszik a miénk és az ellenfél táblája. Ugyanis a tengeri flottának is van vezérkara, és ők is résztvesznek, - ha sokszor láthatatlanul is - a harcokban, illetve ők látják nagyméretben, hogy milyen kiterjedésű a flottamozgás. Ebben a játékban a harcot egy sziget köré helyeztem, a flotta az öbölben, a vezérkar a parton harcol. A Hadműveleti Parancsnokság bunkerekben van elhelyezve repülőtérrrel, kifutópályával és védelmükre légvédelmi ágyúkkal is rendelkezik. Mivel erre történelmi párhuzam is létezik, az 1942-es évben a Midway-i szigeteken folyt hasonló harc az amerikai és a japán flotta között, az amerikaiaknak kettő, a japánoknak 4 anyahajójuk süllyedt el. A harcot az nyeri meg, aki az ellenség minden részét, anyahajóját, csatahajóját, páncéloshajóját, rombolóját, tengeralattjáróját, valamint hadműveleti parancsnokságát, repülőterét, leszállópályáját, és légvédelmi ágyúját, az utolsó szálig megsemmisíti.



A játék hosszú lejáratú, mivel $20 \times 20 = 400$ lövés kell ahhoz, hogy minden mezőt eltaláljunk, bár a gyakorlat azt hozta, hogy 310-315 is elég a játék megnyeréséhez. Hogy némileg kiszámítható legyen az alakzatok elhelyezése, a hajókat CSAK vízszintesen lehet elhelyezni, és hogy NEHÉZ is legyen, megengedtem, hogy az alakzatok, hajók egymáshoz is érjenek, így mégsem lehet kiszámítani előre a célbavett-lőtt formát. A kijelzésnél viszont mindig látszik, hogy milyen típusú hajót, vagy egyéb alakzatot találtunk el, sőt minden hajótípusnál az utolsó darabnál villog egy "1"-es szám, jelezve, hogy ez a kategória már a végénél tart.

A programozás során megoldandó feladatok:

- a játék magyarázata, főmenü elkészítése
- a játémező kis kockáinak, a hajók: anyahajó, csatahajó, páncéloshajó, romboló és tengeralattjáró, valamint a repülőterei leszállópálya és a légvédelmi ágyú UDG-k elkészítése ((felhasználtam az előzőleg (torpedóromboló játékhöz) készített hajóformákat is.))
- A feliratok, táblabeszámolás elkészítése, (először mindkét oldalon számokat tettem, de úgy láttam, hogy az ABC-s koordinátarendszer jobb lesz), a hajók és objektumok bekérése, elraktározása,
- A játék állásának állandó kijelzése, az utolsó találat előtt vészvillogtatás,

- A két tábla állandó váltogatása, gyakorlatilag kétszer ugyanaz a programsorozat valósult meg, az adatok a kétféle DIM számtömbökben tárolódnak, a BASIC lassúságán az emulátor nagyobb sebessége ismét segített,
- Hangjelzések – szirénajelzés, a sikeres találat és a mellément lövések külön-külön kiírása, hangjelzése
- A játék kezdetén a flotta felvonulása, majd egy kis humor, az utolsó kis romboló elkészik a csatába indulóktól, és nagy sebességgel igyekeznek utánuk,
- Lehetőség – a játék végén az ellenfél hadállásának megtekintése,
- Végül, és menet közben többször is, a játék mentése.



Sziréna-szerű riadó hangjelzés, nagybetűre állítjuk a billentyűzetet, a 3000-sor indítja a játékot, letöröljük a képernyőt

```
2 RANDOMIZE USR 60899: POKE 23658,8: GOSUB
3000: CLS
```

az új játéknál véletlenszámra állítás

```
3 RANDOMIZE : GOTO 7
```

amíg nem nyomunk le egy billentyűt, vár és ismétli a ciklust, a keretet feketére, kékre, pirosra, zöldre állítja és villogtatja

```
5 IF INKEY$="" THEN : BORDER 0: BORDER 1:
BORDER 2: BORDER 3: BORDER 4: RANDOMIZE USR
60899: GOTO 5
6 RETURN
```

Sziréna-szerű riadó hangjelzés

```
8 RANDOMIZE USR 60899
```

inicializálás, az alapértékek meghatározása, a változók értéke általában nulla.

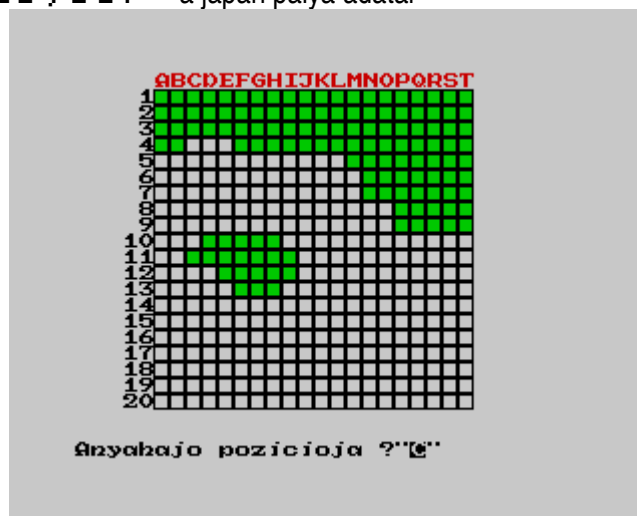
```
10 LET KERES=0: LET GN=8100: LET J=0: LET K=0:
LET INDIT=6: LET HQ=3: LET AGYU=5: LET
TALALAT=0: LET ATT=0: LET EAC=1: LET EB=2: LET
EC=3: LET ED=4: LET ES=5: LET ETALALAT=0: LET
hang=0: LET ch=0: LET INDITX=2: LET AC=1: LET
B=2: LET C=3: LET D=4: LET S=5
15 LET EINDITX=2: LET PSAG=3: LET EINDIT=6:
LET ERA=5: LET EACX=5: LET EBX=8: LET ECX=9: LET
```

```
EDX=8: LET ESX=5: LET ACX=5: LET BX=8: LET CX=9:
LET DX=8: LET SX=5
```



a játékmező tömbjei,itt tárolódnak a hajók és alakzatok koordinátája,mikor felváltva mutatja az amerikai és japán hadállást,innen veszi elő a megfelelő értékeket

```
20 DIM d$(25,30) -az amerikai pálya adatai
25 DIM e$(25,30) -a pályatartozékok adatai
30 DIM f$(25,30) - a japán pálya adatai
```



a 20*20-as pályarácsozat

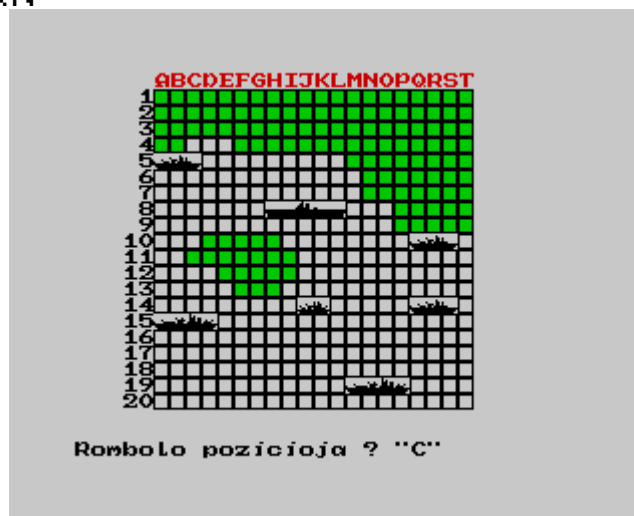
```
50 PAPER 7: BORDER 7: INK 0: CLS : POKE 23692,2:
FOR N=1 TO 21: LET E$(N)=""
XXXXXXXXXXXXXXXXXXXXXXXXX": NEXT N
70 FOR n=1 TO 20: PRINT AT n,5;e$(n,5 TO 25)
74 IF N<10 THEN PRINT AT N,4;N
75 IF N>9 THEN PRINT AT N,3;N
79 NEXT N
```

beszámozzuk a felső oldalát a pályamezőnek

```
80 PRINT INK 2;AT 0,5;"ABCDEFGHJKLMNOPQRST"
```

az 'l\$' string tartalmazza a "sziget" formáját

```
115 GOSUB 120
116 GOTO 200
120 RESTORE 400: READ l$: FOR n=1 TO 3: PRINT
PAPER 4;AT 0+n,5;l$: NEXT n
125 PRINT PAPER 4;AT 4,5;"AA"; PAPER 7;"ABC";
PAPER 4;"XXXXXXXXXXXXXXXXXXXXX "
130 RESTORE 410:: READ l$: PRINT PAPER 4;AT
5,17;l$
140 RESTORE 415: READ l$: PRINT PAPER 4;AT
6,18;l$;AT 7,18;l$
150 RESTORE 420: READ l$: PRINT PAPER 4;AT
8,20;l$;AT 9,20;l$
160 RESTORE 420: READ l$: PRINT PAPER 4;AT
10,8;l$;AT 12,9;l$
165 RESTORE 415: READ l$: PRINT PAPER 4;AT
11,7;l$;AT 13,10;"xxx"
170 RETURN
```



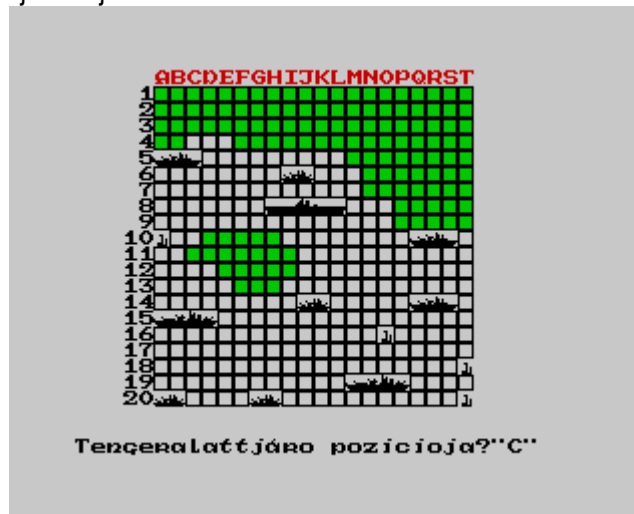
a "szigetek" adatai

```
400 DATA "x"
410 DATA "x"
415 DATA "x"
420 DATA "x"
```

megkezdődik a hajók (anyahajó) adatainak bekérése, korlátozva, hogy csak 1-20 és A-T lehet, majd a megfelelő DATA sorokból meg is rajzolja a hajót

```
600 DIM A$(3): INPUT "Anyahajo pozicioja ?";A$
602 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 600
603 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 600
605 GOSUB 8000
606 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
600
607 FOR N=0 TO 4: IF (E$(X,Y+N) <> "A" OR
ATTR (X,Y+N) <> 56) THEN GOSUB 5500: GOTO 600
608 NEXT N
609 RESTORE 9600: FOR N=0 TO 4: READ Z$: LET
E$(X,Y+N)=Z$: PRINT AT X,Y+N;Z$: NEXT N
610 LET q=0
```


megkezdődik a hajók (csatahajó) adatainak bekérése, korlátozva, hogy csak 1-20 és A-T lehet, majd a megfelelő DATA sorokból meg is rajzolja a hajót



```

611 INPUT "Csatahajó pozíciója? ";A$
612 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 611
613 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 611
614 GOSUB 8000
615 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
611
616 FOR N=0 TO 3: IF (E$(X,Y+N) <> "A" OR
ATTR (X,Y+N) <> 56) THEN GOSUB 5500: GOTO 611
617 NEXT N
618 RESTORE 9610: FOR N=0 TO 3: READ Z$: LET
E$(X,Y+N)=Z$: PRINT AT X,Y+N;Z$: NEXT N
620 LET q=q+1: IF q<2 THEN GOTO 611
629 LET q=0

```

megkezdődik a hajók (páncéloshajó) adatainak bekérése, korlátozva, hogy csak 1-20 és A-T lehet, majd a megfelelő DATA sorokból meg is rajzolja a hajót



```

630 INPUT "Panceloshajó pozíciója ? ";A$
631 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 630

```

```

632 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 630
633 GOSUB 8000
634 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
630
635 FOR N=0 TO 2: IF (E$(X,Y+N) <> "A" OR
ATTR (X,Y+N) <> 56) THEN GOSUB 5500: GOTO 630
636 NEXT N
638 RESTORE 9920: FOR N=0 TO 2: READ Z$: LET
E$(X,Y+N)=Z$: PRINT AT X,Y+N;Z$: NEXT N
640 LET q=q+1: IF q<3 THEN GOTO 630
650 LET q=0

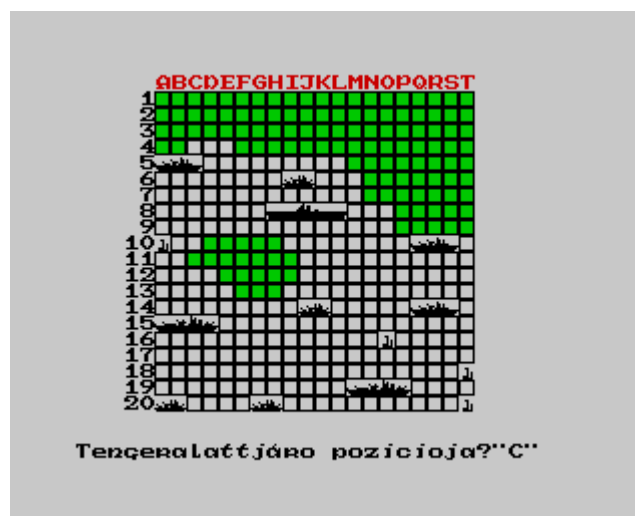
```

megkezdődik a hajók (romboló) adatainak bekérése, korlátozva, hogy csak 1-20 és A-T lehet, majd a megfelelő DATA sorokból meg is rajzolja a hajót

```

655 INPUT "Rombolo pozicioja ? ";A$
656 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 655
657 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 655
658 GOSUB 8000
661 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
655
663 FOR N=0 TO 1: IF (E$(X,Y+N) <> "A" OR
ATTR (X,Y+N) <> 56) THEN GOSUB 5500: GOTO 655
664 NEXT N
665 RESTORE 9930: FOR N=0 TO 1: READ Z$: LET
E$(X,Y+N)=Z$: PRINT AT X,Y+N;Z$: NEXT N
670 LET q=q+1: IF q<4 THEN GOTO 655
695 LET q=0

```

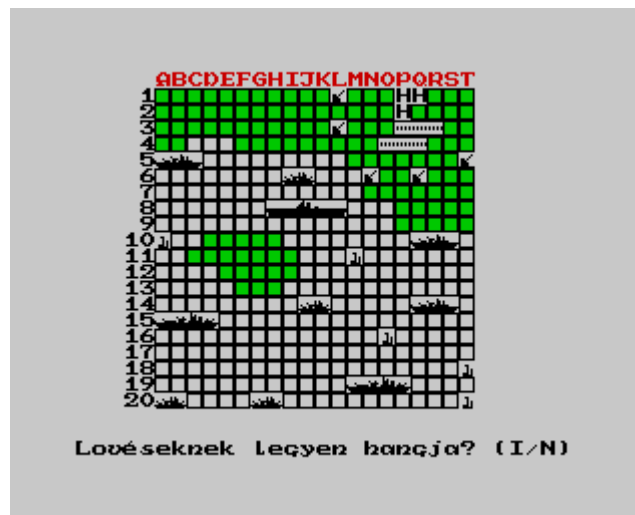


megkezdődik a hajók (tengeralattjáró) adatainak bekérése, korlátozva, hogy csak 1-20 és A-T lehet, majd a megfelelő DATA sorokból meg is rajzolja a hajót

```

700 INPUT "Tengeralattjáró pozíciója?";A$
701 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 700
702 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 700
703 GOSUB 8000
706 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
700
707 IF (E$(X,Y) <> "A" OR ATTR (X,Y) <> 56)
THEN GOSUB 5500: GOTO 700
709 RESTORE 9940: READ Z$: LET E$(X,Y)=Z$:
PRINT AT X,Y;Z$
720 LET q=q+1: IF q<5 THEN GOTO 700
730 FOR N=1 TO 7: PAUSE 10: BORDER N: NEXT N

```



Itt a vezérkar, még nem tudják ,milyen nehéz helyzetbe kerültek

```

750 INPUT "Hadm.p.sag pozíciója ? . ";A$
751 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 750
752 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 750
753 GOSUB 8000
754 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
750
755 FOR N=0 TO 1: IF (E$(X,Y+N) <> "A" OR
ATTR (X,Y+N) <> 32 OR ATTR (X+1,Y) <> 32 OR
E$(X+1,Y) <> "B") THEN GOSUB 5500: GOTO 750
756 NEXT N
758 FOR n=0 TO 1: RESTORE 9950: READ Z$: LET
E$(X,Y+N)=Z$: PRINT AT X,Y+N;Z$: NEXT N: LET
e$(X+1,y)=Z$: PRINT AT X+1,y;Z$
759 LET q=0

```

a repülőtéri kifutópálya

```

760 INPUT "Betonút pozíciója ? ";A$
761 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 760
762 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 760
763 GOSUB 8000

```

```

765 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
754
766 FOR N=0 TO 2: IF (E$(X,Y+N) <> "A" OR
ATTR (X,Y+N) <> 32) THEN GOSUB 5500: GOTO 760
767 NEXT N
768 FOR n=0 TO 2: RESTORE 9960: READ Z$: LET
E$(X,Y+N)=Z$: PRINT AT X,Y+N;Z$: NEXT N
770 LET q=q+1: IF q<2 THEN GOTO 760
779 LET Q=0

```

a parancsnokság védelmét ellátó ágyúk

```

780 INPUT "Legvedelmi agyu poz.:";A$
781 IF (CODE A$(1)>84 OR CODE A$(1)<65) THEN
GOSUB 5500: GOTO 780
782 IF (CODE A$(2)>57 OR CODE A$(1)<48 OR
A$(2)=" " OR CODE A$(3)>57 OR (CODE A$(3)<48 AND
CODE A$(3) <> 32)) THEN GOSUB 5500: GOTO 780
783 GOSUB 8000
785 IF (X>20 OR X<1) THEN GOSUB 5500: GOTO
780
786 IF (E$(X,Y) <> "t" OR ATTR (X,Y) <> 32)
THEN GOSUB 5500: GOTO 780
787 RESTORE 9970: READ Z$: LET E$(X,Y)=Z$:
PRINT AT X,Y;Z$: LET Q=Q+1: IF Q<5 THEN GOTO 780

```



ezt is kellett korlátozni, aki mellette ül csak, elég kellemetlen hallgatnia háborús hangokat

```

788 PRINT #0; "Lovéseknek legyen hangja?
(I/N)": PAUSE 0: INPUT ""
789 IF CODE INKEY#=73 THEN LET hang=1
800 GOTO 6000
810 GOSUB 820: GOSUB 1000: IF KERES=1 THEN
GOTO GN
811 LET R=INT (RND*20)+1: LET M=INT
(RND*20)+5
812 IF E$(R,M)="*" THEN GOTO 811
813 IF E$(R,M)="A" THEN PRINT AT 0,1;" "; AT
1,1;" ": GOTO 811
814 IF (ATTR (R,M)=32 AND hq=0 AND indit=0
AND agyu=0) THEN GOTO 811
815 IF (ATTR (R,M)=56 AND AC=0 AND B=0 AND
c=0 AND D=0 AND S=0) THEN GOTO 811
816 PRINT AT 1,1;R
817 LET pos=60+m: PRINT AT 0,1;(CHR$ pos)
819 GOTO 830

```

a pálya jobboldalán történik az állandó,és folyamatos kijelzés,hogy milyen erők vannak még,illetve semmisültek meg

```

820 PRINT AT 2,26;"AH= ";AC;AT 4,26;"CS=
";B;AT 6,26;"PH= ";C;AT 8,26;"RO= ";D;AT
10,26;"TA= ";S;AT 14,26;"HP= ";H0;AT 15,26;"LA=
";AGYU;AT 16,26;"UT= ";INDITX

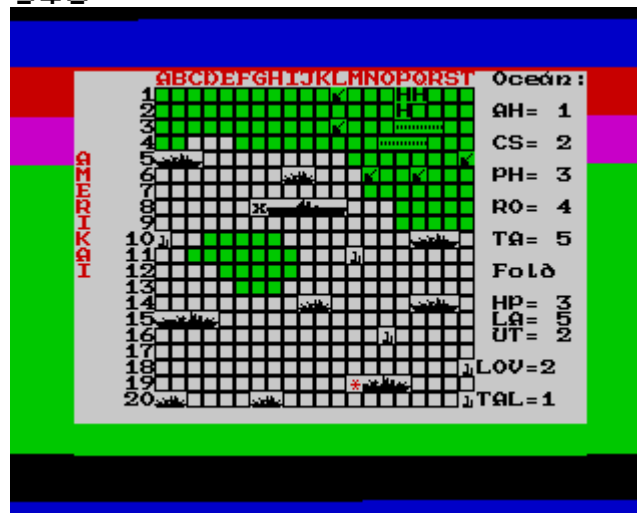
```

külön kijelzés,hogy mi történt a vizen és szárazon

```

825 PRINT AT 0,26;"OCEAN:"
826 PRINT AT 12,26;"FOLD"
827 PRINT AT 20,31;" ";AT
20,25;"TAL=";TALALAT;AT 18,25;"LOU=";ATT
828 RETURN
830 GOSUB 900: GOTO 845
845 IF E$(R,M)="*" THEN PRINT AT 0,1;" ";AT
1,1;" "
896 IF e$(r,m)="A" THEN GOSUB 9990: LET
e$(r,m)="B": PRINT INK 1;AT r,m;"C": PAUSE 30:
GOTO 948
899 GOTO 948

```



ha eltaláltunk egy hajót,hozzáadja a találatjelzőhöz,valamit a táblán is vörösen jelez,majd a keresésben egy újabb kört tesz, minden egységet külön és külön megvizsgál

```

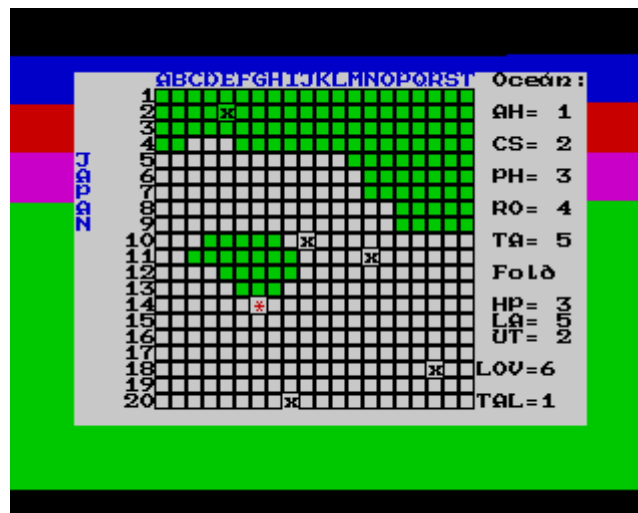
903 IF (E$(R,M)="A" OR e$(r,m)="B" OR
e$(r,m)="C" OR e$(r,m)="D") THEN LET
talalat=talalat+1: GOSUB 9300: LET E$(R,M)="*":
LET BX=BX-1: PRINT INK 2;AT R,M;"*": LET KERES=1
910 IF (E$(R,M)="A" OR e$(r,m)="B" OR
e$(r,m)="C" OR e$(r,m)="D" OR e$(r,m)="E") THEN
LET talalat=talalat+1: GOSUB 9300: LET
E$(R,M)="*": LET ACX=ACX-1: PRINT INK 2;AT
R,M;"*": LET KERES=1
920 IF (E$(R,M)="A" OR e$(r,m)="B" OR
e$(r,m)="C") THEN GOSUB 9300: LET
talalat=talalat+1: LET E$(R,M)="*": LET CX=CX-1:
PRINT INK 2;AT R,M;"*": LET KERES=1
926 IF (E$(R,M)="A" OR e$(r,m)="B") THEN LET
talalat=talalat+1: GOSUB 9300: LET E$(R,M)="*":
LET DX=DX-1: PRINT INK 2;AT R,M;"*": LET KERES=1

```

```

927         IF          E$(R,M)="t"          THEN          LET
talalat=talalat+1: GOSUB 9300: LET E$(R,M)="*":
LET SX= SX-1: PRINT  INK 2; AT R,M; "*" : LET KERES=1
935         IF          E$(R,M)="H"          THEN          LET
talalat=talalat+1: GOSUB 9300: LET E$(R,M)="*":
LET HQ=HQ-1: PRINT  PAPER 8; INK 2; AT R,M; "*" :
LET KERES=1
937         IF          E$(R,M)="t"          THEN          LET
talalat=talalat+1: GOSUB 9300: LET E$(R,M)="*":
LET AGYU=AGYU-1: PRINT  PAPER 8; INK 2; AT
R,M; "*" : LET KERES=1
940         IF          E$(R,M)="t"          THEN          LET
talalat=talalat+1: GOSUB 9300: LET E$(R,M)="*":
LET INDIT=INDIT-1: PRINT  PAPER 8; INK 2; AT
R,M; "*" : LET KERES=1
944         RETURN
948         PRINT AT 0,1;"  "
949         PRINT AT 1,1;"  "
950         LET ATT=ATT+1
960         IF          E$(R,M)="A"          THEN          LET  E$(R,M)="B":
PRINT INK 1; AT R,M; "C"
970         IF (AC=0 AND B=0 AND C=0 AND D=0 AND S=0
AND indit=0 AND agyu=0 AND hq=0) THEN GOTO 1210
999         GOTO 1211

```



ha találtunk,(hajótípusonként és alakzatokként külön-külön hozzáírja),alatta a lövésszámot jelzi,ha az adott típusból már csak egy van,a számot villogtatja (ez aztán a feltételsor !)

```

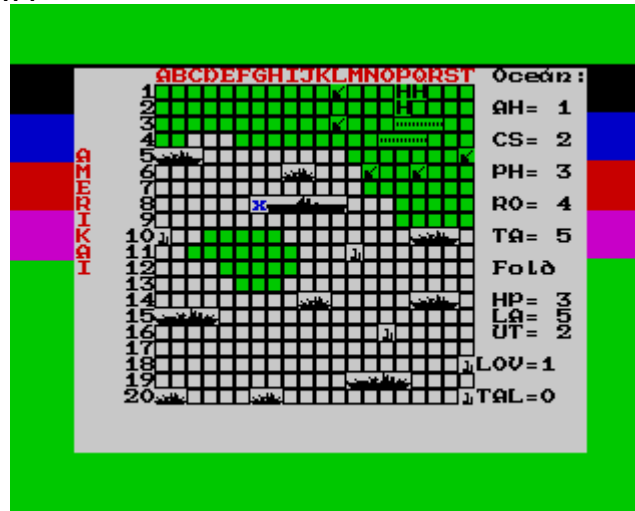
1050        PRINT AT 20,25;"TAL=";TALALAT; AT
18,25;"LOV=";ATT
1100        IF ACX=0 THEN LET AC=0
1101        IF ACX=1 THEN PRINT INK 2; AT 2,30;"1"
1105        IF BX=4 THEN LET B=1
1106        IF b=1 THEN PRINT INK 2; AT 4,30;"1"
1110        IF BX=0 THEN LET B=0
1115        IF CX=6 THEN LET C=2
1120        IF CX=3 THEN LET C=1
1122        IF C=1 THEN PRINT INK 2; AT 6,30;"1"
1125        IF CX=0 THEN LET C=0
1130        IF DX=6 THEN LET D=3
1135        IF DX=4 THEN LET D=2
1136        IF INDIT=3 THEN LET INDITX=1
1137        IF INDIT=0 THEN LET INDITX=0
1140        IF DX=2 THEN LET D=1
1142        IF D=1 THEN PRINT INK 2; AT 8,30;"1"
1145        IF DX=0 THEN LET D=0
1150        LET S=SX
1167        IF S=1 THEN PRINT INK 2; AT 10,30;"1"

```

```

1168 IF HQ=1 THEN PRINT INK 2;AT 14,30;"1"
1169 IF AGYU=1 THEN PRINT INK 2;AT 15,30;"1"
1170 IF SX=0 THEN LET S=0
1171 IF INDITX=1 THEN PRINT INK 2;AT 16,30;"1"
1180 IF AC=0 THEN PRINT AT 2,30;"A"
1185 IF B=0 THEN PRINT AT 4,30;"B"
1190 IF C=0 THEN PRINT AT 6,30;"C"
1195 IF D=0 THEN PRINT AT 8,30;"D"
1199 IF (AC=0 AND B=0 AND C=0 AND D=0 AND S=0
AND indit=0 AND agyu=0 AND hq=0) THEN GOTO 1210
1200 IF S=0 THEN PRINT AT 10,30;"E"
1201 IF HQ=0 THEN PRINT AT 14,30;"F"
1202 IF AGYU=0 THEN PRINT AT 15,30;"G"
1203 IF INDITX=0 THEN PRINT AT 16,30;"H"
1204 RETURN

```

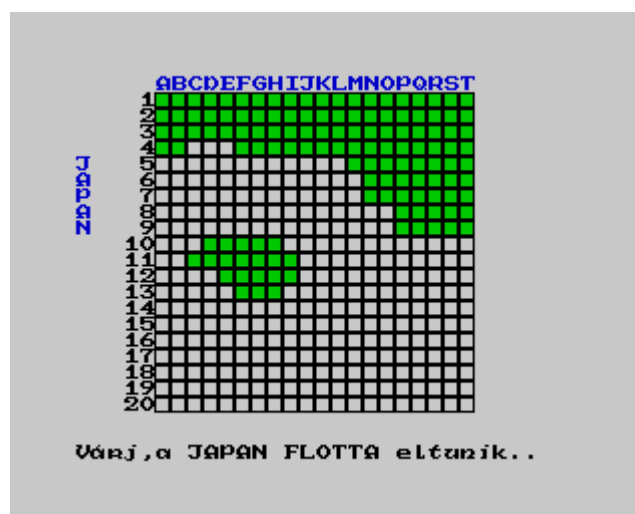


ha minden típusnál nulla van, az amerikaiak elvesztették a csatát

```

1210 PRINT INK 2;#0;"Ha jöhödnak vege !":
PAUSE 200: INPUT "": GOTO 4510
1500 IF (AC=0 AND B=0 AND C=0 AND D=0 AND S=0
AND indit=0 AND agyu=0 AND hq=0) THEN GOTO 1210
2000 GOSUB 820: GOSUB 1000: GOSUB 5:: GOTO
7100
2499 GOTO 2600

```

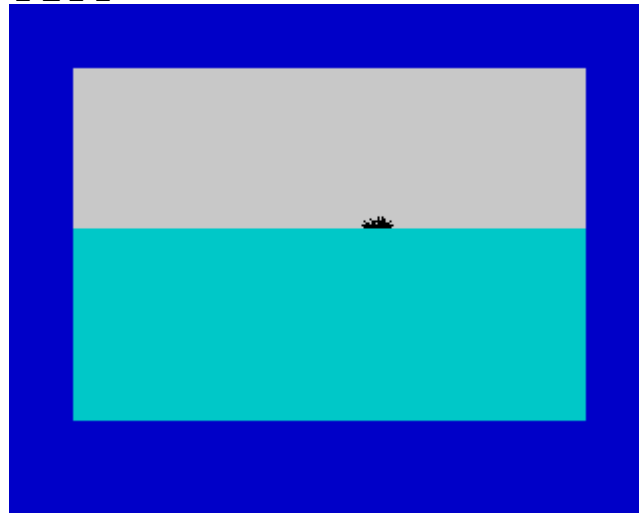


ha viszont itt nulla minden, csak megismételtük a történelmet, újra legyőztük a japánokat

```

2500 GOSUB 5000: PRINT #0;"Gratulalok!
Legyőzted a flottát!": PAUSE 200: INPUT ""
2510 GOTO 4510
2999 GOTO 3100

```



itt kezdődik a játék,rajzos DEMO-val,és egy kis humorral,ugyanis mikor felvonul a hajóhad,a szemléről elkésik az utolsó torpedóromboló és tízszeres sebességgel igyekeznek a többi után

```

3000 BORDER 1: PAPER 7: INK 0: CLS : RANDOMIZE
USR 60899: PRINT INK 1;AT 6,0;"HARC A MIDWAY
SZIGETEK $\square$ RT"
3002 GOTO 4505
3004 GOTO 3100
3010 PRINT #0;" NYOMJ EGY GOMBOT ... ":
PAUSE 0: INPUT ""
3020 BORDER 5: INK 1: PAPER 5: CLS
3025 LET SG=3500
3030 GOSUB SG
3035 GOSUB 5000
3052 LET SG=3600: PRINT #0;"Nyomj egy gombot a
folytat $\square$ shoz": PAUSE 0: INPUT "": CLS : GOTO
3030
3053 PRINT #0;"Nyomj egy gombot a
folytat $\square$ shoz": PAUSE 0: INPUT "": CLS : GOTO 7
3072 PRINT #0;"Nyomj egy gombot Admir $\square$ lis."
3075 PAUSE 0
3080 RETURN

```

a játék magyarázata,hogyan kell majd a koordinátákat megadni

```

3500 CLS : INK 1: PRINT " MIDWAY CSATA
(c) 1998 Nyitrai Laszlo.
A TENGERI CSATA MEGISMETLI A TORTENELMET,UJRA
MEGUIVJUK A HARCOT A HODITO JAPANOK ELLEN."

```




```

3501 INK 2: PRINT " A hadműveletben
resztvevők:"
3502 INK 1: PRINT "xxxxx = Anyahajó
      AH      xxxx = Csatahajó      CS
      xxx    = Panceloshajó      PH
      xx     = Romboló            RO      t
      =      Tengerelattjáró      TA
      HH
      Hadműveleti P.ság      HP      H      =
      Leszállópallya
      UT      = Légvédelmi Ágyú      LA
      & *      Találat      amerikai/japan
      Elhibázott lövés"
3503 INK 2: PRINT " TOUABB
?": PAUSE 0
3505 CLS
3600 PRINT "A flotta felállításához :
Ird be azt a koordinátát, amely az objektum bal
oldali pozíciója
Példa : "
3601 INK 1: PRINT " Az
anyahajót a második sor, kettős mezőjéhez
szeretnénk helyezni, tehát akkor a koordináta a
B2 lesz.
ABCDEFGFG a felállításakor 1ABCD ügyeljünk
arra, 2ABCD hogy a hajókat a 3ABCD
vizbe, a parancsnokságot pedig a szárazföldre
helyezzük, mert hibajelzést kapunk. Amikor a
border villog, nyomjuk az ENTER-t a számítógép
lépéséhez."
3605 GOTO 3053

```

felépítjük a harcmező-táblát, beszámozzuk, fent az ABC betűvel, baloldalt 1-től 20-ig, oldalán amerikai és japán jelzés

```

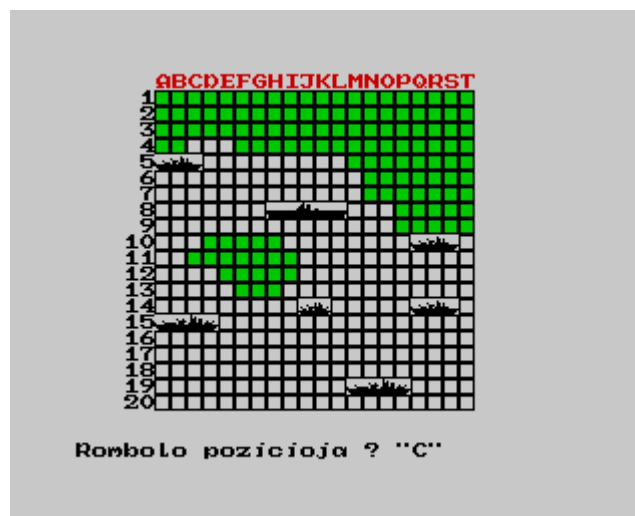
4000 RANDOMIZE : GOSUB 120: FOR n=1 TO 20:
PRINT PAPER 8; AT n,5;e$(n,5 TO 24)
4002 IF n<10 THEN PRINT AT N,4;n
4004 IF n>9 THEN PRINT AT N,3;n
4005 NEXT n
4030 PRINT INK 2; AT 0,4;"
ABCDEFGHIJKLMNPOQRST"
4040 PRINT INK 2; AT
5,0;"A" / "M" / "E" / "R" / "I" / "K" / "A" / "I"
4500 GOTO 810

```

```

4505 FOR N=10 TO 21: PRINT INK 0; PAPER 5; AT
N,0; "": NEXT N:
GOTO 4520
4510 GOSUB 5800: FOR N=10 TO 21: PRINT INK 0;
PAPER 5; AT N,0; "":
NEXT N
4520 RESTORE 9600: FOR n=0 TO 4: READ z$:
PRINT AT 9,27+n; z$: NEXT n
4530 FOR n=26 TO 10 STEP -1: PAUSE 5: PRINT AT
9,n; "AB ": NEXT n
4539 LET Q=0: LET A=0
4540 RESTORE 9610: FOR n=0 TO 3: READ z$:
PRINT PAPER 5; INK 0; AT 10,28+N; z$: NEXT n
4550 FOR n=27 TO 11+A STEP -1: PAUSE 5: PRINT
PAPER 5; INK 0; AT 10,n; "AB ": NEXT n
4552 LET Q=Q+1: LET A=A+5
4555 IF Q<2 THEN GOTO 4540
4579 LET a=0: LET q=0

```



a pályamező kockáit, és a hajókat beolvassa a megfelelő DATA adatokból

```

4580 RESTORE 9620: FOR n=0 TO 2: READ z$:
PRINT PAPER 5; INK 0; AT 11,29+n; z$: NEXT n
4590 FOR n=29 TO 12+a STEP -1: PAUSE 5: PRINT
PAPER 5; INK 0; AT 11,n; "AB ": NEXT n
4592 LET q=q+1: LET a=a+4
4595 IF q<3 THEN GOTO 4580
4639 LET q=0: LET a=0
4640 RESTORE 9930: FOR n=0 TO 1: READ z$:
PRINT PAPER 5; INK 0; AT 12,30+n; z$: NEXT n
4650 FOR n=29 TO 13+a STEP -1: PAUSE 5: PRINT
PAPER 5; INK 0; AT 12,n; "AB ": NEXT n
4662 LET q=q+1: LET a=a+3
4664 IF q<3 THEN GOTO 4650
4699 LET q=0: LET a=0
4700 RESTORE 9940: READ z$: PRINT PAPER 5; INK
0; AT 13,31; z$
4710 FOR n=30 TO 14+a STEP -1: PAUSE 5: PRINT
PAPER 5; INK 0; AT 13,n; "A": NEXT n
4712 LET q=q+1: LET a=a+2
4715 IF q<5 THEN GOTO 4700
4740 DIM D$(116): LET K=1: GOSUB 4990
4750 FOR N=1 TO 4: FOR M=1 TO 31
4760 PRINT AT N,M; D$(K)
4761 LET K=K+1
4769 IF K>116 THEN LET K=1: GOTO 4775
4770 NEXT M: NEXT N

```

utolsó kérdés,hogy merünk-e a hadműveletbe bevágni

```
4775 PRINT #0;"VALASZ ?(I/N)"; PAUSE 0: INPUT  
""  
4780 IF INKEY$ <> "I" THEN GOTO 9495  
4781 FOR N=0 TO 4: PRINT AT N,0;"  
": NEXT N  
4790 GOSUB 5000: FOR N=0 TO 200: RANDOMIZE USR  
64243: NEXT N  
4800 FOR N=29 TO 0 STEP -1: PAUSE 2: BEEP  
.01,.01: PRINT AT 9,N;"AB ": NEXT N  
4805 PRINT AT 9,0;" "  
4810 GOSUB 3500  
4989 INDIT 3  
4990 LET D$=" A FLOTTA MOST HARCRAKESZ  
MEGVEDJUK A SZIGETEKET A HODITO JAPAN  
KATONAKTOL? "  
4992 RETURN
```



megszólal a harci sziréna,riadóval indul

```
5000 FOR n=1 TO 3  
5001 RANDOMIZE USR 64647  
5002 PAUSE 5  
5003 NEXT n  
5099 RETURN  
5100 FOR n=5 TO 1 STEP -.1  
5101 BEEP .01,n  
5102 NEXT n  
5103 BEEP .6,.1  
5110 RETURN
```

ha a koordinátákban - bevitel közben hiba van – figyelmeztetés (főleg az szokott előfordulni,hogy valamelyik hajót mindenáron a szárazföldre akarjuk tenni)

```
5500 PRINT INK 2;#0;"BEVITELI HIBA, KERLEK  
ISMETELD": PAUSE 0: INPUT "": RETURN
```

mikor vége a játéknak,a számítógép megmutatja az ellenfél elhelyezkedését,mit is kellett volna eltalálnunk

```

5800 POKE 23659,0: PRINT AT 22,0;"NYOMD
""R""MEGMUTATOM A JAPAN FLOTTAT, VAGY BARMI MAS
FOLYTAT ": PAUSE 1: PAUSE 1: PAUSE 0: PRINT AT
22,0;"
": POKE 23659,2
5810 IF CODE INKEY# <> 82 THEN INPUT "": CLS :
RETURN
5815 GOSUB 120: FOR n=1 TO 20: PRINT PAPER
8; AT n,5;d$(n,5 TO 24): NEXT n
5820 FOR n=1 TO 20: FOR m=5 TO 24: IF (f$(n,m)
<> "B" AND f$(n,m) <> "C" AND f$(n,m) <> "C")
THEN PRINT AT n,m;f$(n,m)
5830 NEXT m: NEXT n

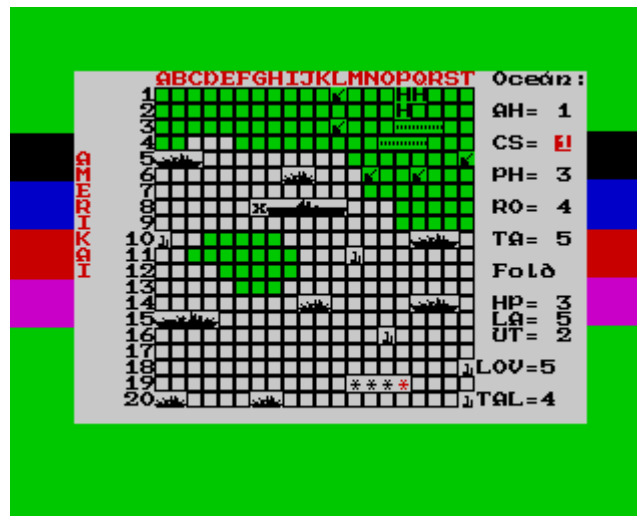
```

új játék indítása, ez általában elmarad, az előző is nagyon fárasztó volt

```

5840 PRINT #0;"FOLYTATASHOZ EGY GOMB NYOMAS":
PAUSE 0: CLS
5850 RETURN
6000 CLS : RANDOMIZE : POKE 23692,2: FOR N=1
TO 21: LET F$(N)="UUUUUUUUUUUUUUUUUU": NEXT N
6005 CLS : POKE 23692,2: FOR N=1 TO 21: LET
d$(N)="UUUUUUUUUUUU": NEXT N
6010 FOR n=1 TO 20: PRINT AT N,5;D$(n,5 TO 25)
6015 IF N<10 THEN PRINT AT n,4;n: NEXT n
6017 IF N>9 THEN PRINT AT N,3;N: NEXT N
6019 NEXT N

```



amikor átvált a japán hadállásra, itt tudunk koordinátákat löni, mutatja az eddigi eredményt

```

6020 PRINT INK 1; AT
0,5;"ABCDEFGHIJKLMNQRST"

```

a JAPAN szó függőleges kiírása

```

6030 PRINT INK 1; AT 5,0;"J""A""P""A""N"
6060 GOSUB 120

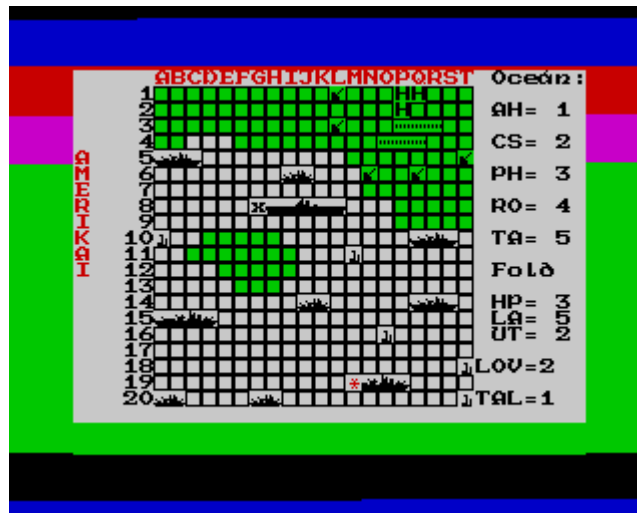
```

amikor a gép bekódolja magát, és a memóriájában elrejt az ellenséget, a számítógép véletlenszám generálással, o-p vízszintes és függőleges pozícióval állítja be a japán flottát, minden pozíciókeresésnél újra és újra a 7000-es sorra ugrik

```

6070 PRINT INK 1;#0;"VARJ,A JAPAN FLOTTA
ELTUNIK.. "
6100 GOSUB 7000
6110 FOR n=1 TO 5: IF (f$(p,o+n) <> "A" OR
ATTR (p,o+n) <> 56) THEN GOTO 6100
6115 NEXT n
6120 RESTORE 9600: FOR n=1 TO 5: READ z$: LET
f$(p,o+n)=z$: NEXT n
6149 LET q=0
6150 GOSUB 7000
6160 FOR n=1 TO 4: IF (f$(p,o+n) <> "A" OR
ATTR (p,o+n) <> 56) THEN GOTO 6150
6165 NEXT n
6170 RESTORE 9610: FOR n=1 TO 4: READ z$: LET
f$(p,o+n)=z$: NEXT n
6172 LET q=q+1: IF q<2 THEN GOTO 6150
6200 LET q=0
6210 GOSUB 7000
6220 FOR n=1 TO 3: IF (f$(p,o+n) <> "A" OR
ATTR (p,o+n) <> 56) THEN GOTO 6210
6225 NEXT n
6230 RESTORE 9620: FOR n=1 TO 3: READ z$: LET
f$(p,o+n)=z$: NEXT n
6235 LET q=q+1: IF q<3 THEN GOTO 6210
6399 LET q=0
6400 GOSUB 7000
6410 FOR n=1 TO 2: IF (f$(p,o+n) <> "A" OR
ATTR (p,o+n) <> 56) THEN GOTO 6400
6415 NEXT n
6420 RESTORE 9930: FOR n=1 TO 2: READ z$: LET
f$(p,o+n)=z$: NEXT n
6425 LET q=q+1: IF q<4 THEN GOTO 6400
6599 LET q=0
6600 GOSUB 7000
6610 LET n=1: IF (f$(p,o+n) <> "A" OR ATTR
(p,o+n) <> 56) THEN GOTO 6600
6620 RESTORE 9940: READ z$: LET f$(p,o+1)=z$
6625 LET q=q+1: IF q<5 THEN GOTO 6600
6830 GOSUB 7000
6840 FOR N=1 TO 2: IF (F$(p,o+n) <> "A" OR
ATTR (P,O+N) <> 32) THEN GOTO 6830
6842 NEXT n
6845 IF (f$(p+1,o+1) <> "A" OR ATTR (P+1,O+1)
<> 32) THEN GOTO 6830
6850 FOR N=1 TO 2: RESTORE 9950: READ Z$: LET
F$(p,o+n)=Z$: LET f$(p+1,o+1)=Z$: NEXT N
6859 LET q=0
6860 GOSUB 7000
6865 IF 0>21 THEN GOTO 6860
6870 FOR N=1 TO 3: IF F$(p,o+n) <> "t" THEN
GOTO 6860
6872 NEXT n
6875 FOR N=1 TO 3: IF ATTR (p,o+n) <> 32 THEN
GOTO 6860
6876 NEXT n
6880 FOR n=1 TO 3: RESTORE 9960: READ Z$: LET
F$(p,o+n)=Z$: NEXT n
6882 LET q=q+1: IF q<2 THEN GOTO 6860
6885 LET q=0
6890 GOSUB 7000
6900 IF (F$(p,o) <> "A" OR ATTR (p,o) <> 32)
THEN GOTO 6890
6910 RESTORE 9970: READ Z$: LET F$(p,o)=Z$
6915 LET q=q+1
6920 IF q<5 THEN GOTO 6890
6999 CLS : GOTO 4000

```



a számítógép véletlenszám generálással céloz és (általában) nem talál,o-p vízszintes és függőleges pozíció

```

7000 LET o=INT (RND*20)+5
7010 LET p=INT (RND*20)+1: RETURN
7100 RANDOMIZE : GOSUB 120: FOR n=1 TO 20:
PRINT PAPER 8;AT n,5;D$(n,5 TO 24)
7110 IF n<10 THEN PRINT AT N,4;n
7120 IF n>9 THEN PRINT AT N,3;N
7136 NEXT N: PRINT AT 0,0;

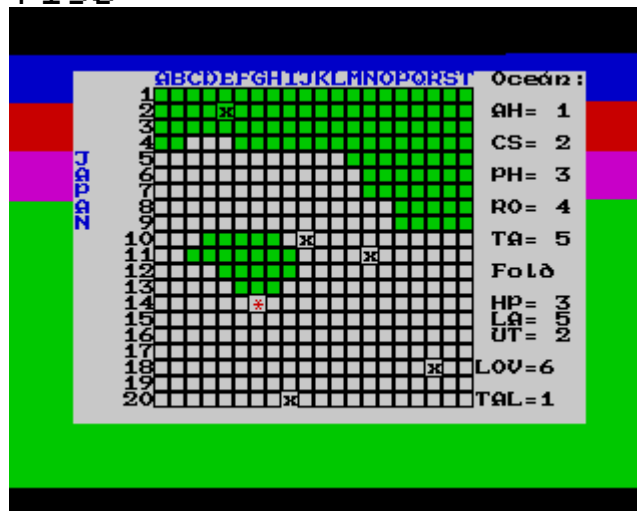
```

újra a "JAPAN" tábla a képernyőn

```

7140 PRINT INK 1;AT 0,5;"ABCDEFGHIJKLMNQRST"
7150 PRINT INK 1;AT 5,0;"J"|"A"|"P"|"A"|"N"|"
7160 GOSUB 7170
7165 GOTO 7190

```



az ellenséges oldalon a találatok jelzése

```

7170 PRINT AT 2,20;"AH = ";EAC
7172 PRINT AT 4,20;"CS = ";EAB
7174 PRINT AT 6,20;"PH = ";EAC
7176 PRINT AT 8,20;"RO = ";EDC
7178 PRINT AT 10,20;"TA = ";E6C
7179 PRINT AT 14,20;"HP = ";P6AG
7180 PRINT AT 15,20;"LA = ";EAA

```

```

7182 PRINT AT 16,26;"UT="";EINDITX
7183 PRINT AT 0,26;"OCEAN"
7184 PRINT AT 12,26;"FOLD"
7187 PRINT AT 18,25;"LOV=";ATT
7188 PRINT AT 20,31;" ";AT
20,25;"TAL=";ETALALAT
7189 GOSUB 7550: RETURN

```

a játékos választhat,vagy pontos koordinátával céloz,vagy a gép segítségével "véletlenül" lövöldöz (ezt igénybe is lehet venni,mert a vége felé már eléggé fárasztó a játék)

```

7190 POKE 23659,0: PRINT AT 22,0;" ENTER-t,ha
celzol, barmi mas gombot,ha
veletlensz.": PAUSE 1: PAUSE 1: PAUSE 0: PRINT
AT 22,0;"
": POKE 23659,2

```

vár addig,amig az "ENTER"-t le nem nyomjuk

```

7191 IF INKEY$=CHR$ 13 THEN GOTO 7200
7192 INPUT ""
7193 IF ch >= 1000 THEN GOTO 7200
7194 LET k=INT (RND*20)+5: LET j=INT
(RND*20)+1
7195 IF (D$(j,k)="A" OR D$(j,k)="B") THEN LET
ch=ch+1: GOTO 7193
7198 GOTO 7299

```

ha olyan pontot lőttünk,mely már volt,hibajelzés,megismételhetjük

```

7199 PRINT #0;"Ismeteld!": PAUSE 0: INPUT ""
7200 DIM K$(3): INPUT "Ird be a
koordinatat. ";k$

```

ez a sor segít ,ha már nem találtunk meg semmit,ha három nullát (CHR\$ 48)beirunk,akkor az 5800-as soron lévő kérdésre ugrik,megmutatja nekünk a JAPÁN hadállást

```

7203 IF k$=CHR$ 48+ CHR$ 48+ CHR$ 48 THEN GOTO
5800

```

korlátozva,hogy csak a megfelelő (memóriában lévő)értéket adjunk be

```

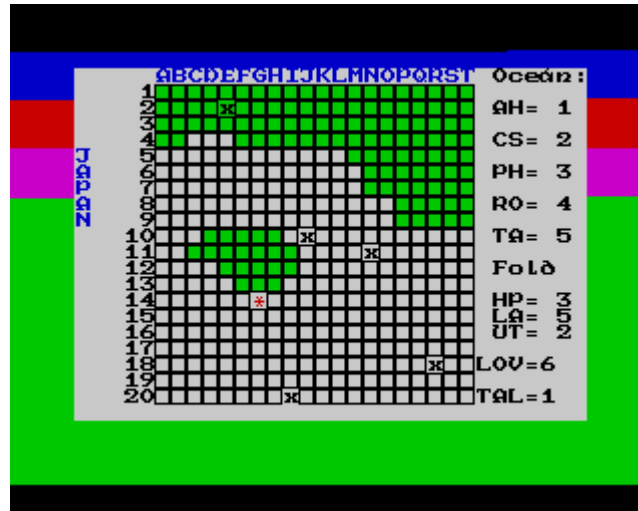
7205 LET K=(CODE K$(1)-60)
7210 IF (K<5 OR K>25) THEN GOTO 7200
7240 IF (CODE K$(2)>57 OR CODE K$(1)<48 OR
K$(2)=" " OR CODE K$(3)>57 OR (CODE K$(3)<48 AND
CODE K$(3) <> 32)) THEN GOTO 7200
7250 LET J=VAL (K$(2 TO 3))
7280 IF (J<1 OR J>20) THEN GOSUB 5500: GOTO
7200

```

```

7300 IF f$(j,k)="A" THEN GOSUB 9990: LET
d$(j,k)="B": PRINT AT j,k;"C": FOR n=1 TO 50:
NEXT n: LET ch=0: GOSUB 5: GOTO 4000
7303 IF (d$(j,k)="A" AND CH=0) THEN GOTO 7199
7304 IF (d$(j,k)="B" AND CH>0) THEN GOTO 7193
7305 IF (d$(j,k)="C" AND CH=0) THEN GOTO 7199
7306 IF (d$(j,k)="D" AND CH>0) THEN GOTO 7193

```



találatok után az objektum száma eggyel csökken

```

7310 IF (f$(j,k)="A" OR f$(j,k)="B" OR
f$(j,k)="C" OR f$(j,k)="D") THEN LET EBX=EBX-1
7330 IF (f$(j,k)="B" OR f$(j,k)="A" OR
f$(j,k)="C" OR f$(j,k)="B" OR f$(j,k)="D") THEN
LET EACX=EACX-1
7340 IF (f$(j,k)="C" OR f$(j,k)="A" OR
f$(j,k)="B") THEN LET ECX=ECX-1
7370 IF f$(j,k)="D" THEN LET ES=ES-1
7375 IF (f$(j,k)="E" OR f$(j,k)="D") THEN LET
EDX=EDX-1
7385 IF f$(j,k)="F" THEN LET EINDIT=EINDIT-1
7390 IF f$(j,k)="G" THEN LET EAA=EAA-1
7395 IF f$(j,k)="H" THEN LET PSAG=PSAG-1
7396 IF hang=1 THEN GOSUB 5100

```

a lövések után a találatszám mindig nő

```

7398 LET etalalalat=etalalalat+1
7400 LET d$(j,k)="A": PRINT AT j,k;"B"
7410 IF etalalalat=49 THEN GOTO 7980
7450 IF ETALALAT=49 THEN GOTO 2500
7500 IF EACX=0 THEN LET EAC=0
7505 IF EBX=4 THEN LET EB=1
7510 IF EBX=0 THEN LET EB=0
7515 IF ECX=6 THEN LET EC=2
7520 IF ECX=3 THEN LET EC=1
7525 IF ECX=0 THEN LET EC=0
7530 IF EDX=6 THEN LET ED=3
7535 IF EDX=4 THEN LET ED=2
7540 IF EDX=2 THEN LET ED=1
7542 IF EINDITX=3 THEN LET EINDIT=1
7543 IF EINDITX=0 THEN LET EINDIT=0
7545 IF EDX=0 THEN LET ED=0

```



```

7547 GOSUB 7550
7549 GOTO 7593

```

a találatok után,ha már csak egy van az adott hajóból,vagy objektumból,kijelzés villogtatva

```

7550 IF EACX=1 THEN PRINT AT 2,30;"1"
7552 IF EACX=0 THEN PRINT AT 2,30;"A"
7555 IF EB=1 THEN PRINT AT 4,30;"1"
7557 IF EB=0 THEN PRINT AT 4,30;"B"
7560 IF EC=1 THEN PRINT AT 6,30;"1"
7562 IF EC=0 THEN PRINT AT 6,30;"C"
7565 IF ED=1 THEN PRINT AT 8,30;"1"
7567 IF ED=0 THEN PRINT AT 8,30;"D"
7570 IF ES=1 THEN PRINT AT 10,30;"1"
7572 IF ES=0 THEN PRINT AT 10,30;"E"
7575 IF EINDIT <= 3 THEN PRINT AT 16,30;"1"
7577 IF EINDIT=0 THEN PRINT AT 16,30;"F"
7580 IF EAA=1 THEN PRINT AT 15,30;"1"
7582 IF EAA=0 THEN PRINT AT 15,30;"G"
7585 IF PSAG=1 THEN PRINT AT 14,30;"1"
7587 IF PSAG=0 THEN PRINT AT 14,30;"H"
7590 RETURN
7595 GOSUB 7170: GOSUB 5: GOTO 4000

```

a japánok megváltoztatták a történelmet,most az amerikaiak vesztek

```

7980 GOSUB 5000: PRINT INK 2;#0;"Vege,a
f lottad elsullgyedt.": PAUSE 200: INPUT ""
7990 GOTO 2510
7999 GOTO 9000
8000 LET Y=(CODE A#)-60
8050 LET X=VAL (A$(2 TO 3))
8055 RETURN
8100 LET hy=m: LET q=1
8108 IF (q=-1 AND e$(r,m)="A") THEN GOTO 8114
8109 IF (q=1 AND e$(r,m)="B") THEN LET q=-1:
LET M=HY
8110 IF (M+q >= 5 AND e$(r,M+q) <> "A" AND
e$(r,M+q) <> "*" AND M+q <= 24) THEN LET m=m+q:
GOSUB 900: LET gn=8105: GOTO 845
8114 IF q=-1 THEN LET keres=0: LET gn=8100:
GOTO 811
8115 LET Q=-1: LET M=HY
8130 GOTO 8105
9085 GOTO 6000
9300 IF hang=1 THEN GOSUB 5100
9305 RETURN
9495 CLS : STOP

```



innen veszik a stringek a "hajó" alkatrészeket, ebből építik fel a hajókat, ezek saját készítésű "UDG"-k

```

0000 DATA "A", "B", "C", "D", "E"
0010 DATA "B", "C", "D", "E"
0020 DATA "D", "E", "F"
0030 DATA "A", "G"
0040 DATA "A"
0050 DATA "H"
0060 DATA "D"
0070 DATA "C"
0080 GOTO 9999

```

a vereség után egyre mélyülő ,majd magasodó sziréna - hang -

```

9990 IF hang=1 THEN FOR N=60 TO -60 STEP -10:
BEEP .02,N: NEXT N: FOR N=1 TO 20 STEP 10: BEEP
.01,N: NEXT N
9998 RETURN

```

és végül a játék mentése

```

9999 SAVE "MIDWAY" LINE 3000

```

FÜGGELÉK

* I am Laszlo Nyitrai wery old spectrum programmer *

Short CV:

=====

I was started the computer programming in 1982 with ZX81. I was working as a police officer in the in the Hungarian Ministry of Interior, when I decided to be in an informatics. Graduated in 1985 continued working as a system and program developer still in the Ministry. I was making the up to now used criminal database program. Having IBM 4000/6000 machines at the workplace, I was started as a programmer on my ZX Spectrum 48K+ at home.

Publications:

=====

My ZX Spectrum programs gets wide publicity - and not only in Hungary. Sinclair User were published also my programs in the U.K. Modern press is still publish my works: the Chip magazine, and the Hungarian version(Computer Panorama) of the German Computer Personlich also giving place for my artilces.

Works:

=====

Written about 1000 programs in the last 20 years, meaning very good experience of Zilog programming. It is really would be too long for detail them, but let's take some for example.

My favourites are the table/logical games, but written some arcade games also, as well some another type of programs.

Some examples:

=====

- Awari-Awale: classic table game from the ancient ages
- Backgammon 128K: another table game, extended with the 128K Speccy's possibilities
- Biorhythm: the Speccy version of the classic biorhythm program
- Chessmaster: 6 levels chess program with snell-parties
- Dama: another classic game
- Dori 128K: Manic Miner-like arcade
- Fontmaker: create special fonts, which are useable in your programs.
- Formula 1.: database of the Formula 1., making with my extended Nyitrai 128K

Basic

- Garden: arcade game for kids
- Hopp-Hopi: a Jumping Jack clone
- Labyrinth: find the way out of labyrinth
- Matrix: mathematical logical game with matrix
- Midway: the modern version of the well-known torpedo game
- Othello: another well-known game rewritten in 2002.
- Smurfs: arcade/adventure game for kids, idea taken from the famous cartoon
- Sokoban: classic computer game, with lots of levels and fully animated figure

I am waiting your response, and wishing the bests:

Laszlo Nyitrai
Hungary - Budapest
email: lnyitrai@freemail.hu

Nyitrai László régi spectrum programozó.

=====

1982-ben már 38 éves voltam !!!, mikor egy Sinclair User újság akadt a kezembe és mikor a benne lévő ZX80-as és ZX81-es programokat néztem, elhatároztam, hogy pályát módosítok, megváltoztatom eddigi életem, és én is számítógépes leszek. 1982-ben beiratkoztam a Bánki Donáth Műszaki Főiskola rendszerszervező tagozatára, ahol 1885-ben, mint műszaki tanár -rendszerszervező-programozó-végeztem. Otthagyva előző munkámat, programozó és rendszerfejlesztő lettem az IBM 4000/6000-es nagygépeken.. Ekkor már több kis BASIC programot írtam ZX81-re, de az első igazán érdekes többfordulós SPECTRUM BASIC programot 1984-ben írtam, melyet el is küldtem a Sinclair User-nek, ahol meg is jelent 1984 októberi számában a 91-ik oldalon, RIGHT STUFF néven. A program érdekessége, hogy erről én soha pénzt, értesítést, elismerést nem kaptam, a megjelenésről is csak 1997-ben értesültem, mikor láttam e lap eredetijét. A program alatt csak annyi áll, hogy a szerzőjét nem ismerik és jelentkezzen aki tudja hol van. (Pedig 10 Font járt volna a megjelent programért!!) Ekkor a ZX81 után már rendelkezem egy SPECTRUM + 48k-s keménybillentyűs géppel, majd botoran 1991-ben eladtam és XT-re, majd AT-re tértem át, melyeket aztán szépen cserélgettem 386-486-os, majd a mai gépekre, jelenleg egy Pentium 4-es, 512 MB ramos 2,4 GHZ-es szupergyors gépem van WINXP-vel. Erről a gépről értekezem az Interneten keresztül. 1995-ben ismét a SPECTRUM programozás felé fordultam. Sikerült az eladott Spectrum+ gépemet megtalálni és visszavásárolni a vele adott könyvekkel, többszáz kazettával együtt. Ezeket aztán éveken át-keresztül !!! másolgattam az emulátoromba, míg rájöttem, hogy az Interneten sokkal egyszerűbben hozzájutok ezekhez a programokhoz. Sajnos sok, akkor készített programomat elvesztettem, bár sikerült talán ezekből kb. kétszáz saját programot megmenteni, ezekből élek mai is. Kedvenceim a logikai, táblás játékok, ezeket újra és újra előveszem és módosítok rajtuk. Bár ez a gépem még működőképes, szépen becsomagolva vár a feltámadásra a szekrényben, inkább a kis IBM notebook-om használom, TFT-s színes, Pentium I-es 166 MHz-s, 48 MB ramos, DOS 6.22 alól használom és a kedvenc Gerton Lunter féle Z80-as vagy X128-as emulátorommal dolgozom. Ez nagyszerűen megfelel a célnak, igazi sinclair-es környezetet biztosít. Jelenleg van egy Ebay-en vásárolt Spectrum 128K+2-es gépem is.