

CSUPA CSUPA játék



harmadik

játékgyáros rész
2005

MORE EBOOKS

>49 type-ins from Hungary

Is this the dawn of a new era in type-in listings? Here we have not one, but two whole PDF books crammed full of listings - 49 of them between the two volumes. There's not much I can say about the programs since the program text and annotations are all in Hungarian, although it's certainly worth pointing out that the download for each book contains all of the games typed in for you and saved as Z80 snapshot files as well as the main PDF itself.

László Nyitrai is the author of the two books, totalling nearly 300 pages between them. They can be downloaded from the new books area of WoS's ftp site:

<ftp://ftp.worldofspectrum.org/pub/sinclair/books/>



(az előző könyvek ismertetője megjelent a ZXFormat magazinban 2004/08 számban !!!)

T A R T A L O M

Képes Gábor (KGB) bevezetője

1. KIRÁLYKISASSZONY KATONÁJA
2. FEKETE HATTYÚ
3. RÓKA ÉS LIBÁK (LUDAK ZX81)
4. SÁRGA TENGERALATTJÁRÓ
5. PIRAMIS SZOLITER
6. LÁDATOLOGATÓ (SOKOBAN)
7. MASTERMIND 128K
8. PUZZLE 2004
9. SZÓKERESŐ
10. ÚJ-TÉGLATÖRŐ
11. MAGYAR ZÁSZLÓ
12. HARMADIK ZENEDEMŐ

FÜGGELÉK:

13. SZÖVEGES KALANDJÁTÉK KÉSZÍTÉSE
14. JÁTÉKTERVEZÉS ÚTMUTATÓ

BEVEZETŐ

Számítógépes játék - **Képes Gábor** – tollából

Az informatika - bár egyes teoretikusai szerint egyidős az emberiséggel, lévén, hogy az embert emberré tevő munka, kooperáció és kommunikáció története egyben az információ, az adatmozgatás története is – alig hatvanéves tudomány. A modern számítástechnika alapjait a német Konrad Zuse, a magyar származású Neumann János és kortársaik rakták le, s az első digitális elektronikus számítógépek (köztük legfontosabbként az első Neumann-elvű gép, az IAS) a második világháború utáni években kerültek üzembe.

A számítástechnika, mely eleinte a népszámlálás, a hadtudomány (légvédelem, löelemképzés), a kódfejtés, majd sok évtizedig az ipari vezérlés (innen a kibernetika, irányítástechnika szó) és a magas szintű matematikai-fizikai kutatások szolgálóleánya volt, ezekben az időkben zárt, elit szaktudomány lehetett csak, sőt néhány évig – gyakorlatilag Sztálin haláláig - a világ keleti felén veszélyes polgári elhajlásnak tekintették. A géppark az egész világon is mindössze pár tucat komputert jelentett, s ez a szám a hatvanas évtized végéig is csak néhány ezerre növekedett, bár ez esetben a "csak" szó kizárólag utólagos okoskodásoknál adekvát.

Bizony, a számítástechnika első évtizedei (a mikroprocesszor megjelenéséig) csak utólag tűnhetnek sétagaloppnak, lassú fejlődésnek -egészen elképesztő mérvű fejlődést élt át az a mérnök, aki ilyen gépekközött dolgozhatott. Ezeknek a berendezéseknek a tudása a mai otthoni változataik töredéke volt, az áruk viszont akár a milliószorosa is lehetett. Félelmetes belegondolni, hogy egy ilyen készülék egy holdra szállás hibátlan vezérlésére vagy egy atomerőmű folyamatainak koordinálására is alkalmas volt, könyvtárnyi anyagokat tárolt és rendszerezett – tehát legalább minderre mai asztali gépünk is képes lenne, ha akarnánk rá használni.

De a gépek hatékonyságnövekedése és a felhasználás körei cseppet sem egyenes arányban fejlődtek, sőt! valamiféle deficit alakult ki a hányavetin egymásra fércelt fejlesztések nyomán. Lassan szinte csak játszani van kedvünk egy olyan gépen, amely korábban a legtökéletesebb emberi munka eszköze volt. A gépi agy, amely máig sem váltotta be minden ígérését (itt elsősorban a hibátlan gépi fordítás problematikájára gondolok), tökéletesen visszatükrözi a társadalmat, amelyben működik.

A modernista, haladáselvű és legalább még elemeiben optimista háború utáni társadalom (ez esetben tökéletesen mindegy, hogy keletről vagy nyugatról beszélünk) átlagos számítógépe egy steril környezetben működő szuperagy. A hetvenes-nyolcvanas évek átlagos számítógépe már metszetbe állítja a munka és a játék köreit, de ezek között még világos hierarchia működik. A számítógép az értelmes időtöltés eszköze, amely, mint egy jó tanár, akár képességfejlesztő játékokban is partnerünk lehet.

Posztmodern jelenünkben a játék és a munka körei veszélyesen és játékosan összerosódnak, s így a számítógép világa is képlékeny, örökké változó és veszélyesen ígéretes avagy ígéretesen veszélyes, ahogy tetszik.

Hiszem, hogy nemcsak a hegeli logika szerint, de a folyamatok elemzése is azt mutatják, a posztmodern után egy neomodern következik. Ahogy az építészetben a BAUHAUS-t és a funkcionalizmust felváltó csiricsaré posztmodernt egy a BAUHAUS-t légiességgel és szabadság-igénnyel fellazító (de azt mégiscsak rehabilitáló) neomodern irányzat váltotta fel (ugyanaz újabban a költészetben is megfigyelhető), valószínűleg az élet egyéb területein is felmerül majd a világ egyben látása iránti igény, a visszatérés a közös víziókhöz és a haladásba vetett hithez, amelynek nemléte, bár a bölcséleti tudományokat cseppet sem zavarja, hosszabb távon a legfontosabb kutatási irányokat (orvostudomány, úrkutatás, informatika) véglegesen gúzsba köthetné. De ez, a modernitás reneszánsza talán a mai asztali számítógép halála is lesz: teljesen összerosódik majd telefon, Hi-Fi, televízió, teletext, PC és háztartási eszköz, s az elektronika csak ugyanolyan vívőanyag lesz, mint a levegő.

Ily módon a számítógépes szubkultúra is eltűnik.

Történeti aspektusból a számítástechnika legragyogóbb korszaka az 1975 és 93 közötti időszak volt. Példának okáért ebben a korszakban (1982 és 92 között) csak a Commodore-64- ből húszmillió példányt adtak el. A számítógép berobbant a köztudatba, elindított egy nagyszabású és az élet minden területére

kiterjedő folyamatot (három dimenziós alkalmazások, villámgyors távkommunikáció, új típusú szövegkezelés, virtuális valóság), amelynek az esetleges agresszió és más negatív jelenségek csak sajnálatos mellékhatásai, s amelynek a számítógép (azaz a memóriából, vezérlőműből, számológépből álló, tárolt programú berendezés) újra csak szolgálóleánya lesz, mint volt egykor a - lássuk be, sokkal zordabb - löelemképzésnek.

Hogy a forradalomban katalizáló szerepe van az emberi tényezőnek, az ember-gép kapcsolat metamorfózisainak, az nem kérdés. És hogy az ember-gép kapcsolat legbiztosabb mérőműszere és egyben belső katalizátora a játék, az is tagadhatatlan.

Az első számítógépes játék valószínűleg még maga a hardver volt.

Magyarország első középiskolai számítástechnika tanára, a Piarista Gimnáziumban oktató szerzetes-tanár, Kovács Mihály is a rádióamatőr mozgalom mintájára, klubkeretek között kezdte a gyerekekkel megszerettetni a számítógépet (1959-től!), s ennek a "megszerettetésnek" természetesen a legősibb pedagógiai módszere a játék volt. Készítsünk kibernetikai játékokat, mondta Kovács Mihály a klimatizált termekbe zárt, fegyveres őrökkel védett komputeres korában. Így születtek meg az első magyar számítógépes játékok, a labirintusban szaladgáló műegér, a malmozó- és kártyázógép, majd a programozást játékká tevő Mikromat kibernetikai építőkészlet.

Ezek a játékok építőjátékok voltak, mint az építőkockázás. Más jellegű építőközből, így relékből, zseblámpaizzóból, telepől, huzalokból, de mégiscsak modell- jellegű játékszerek voltak készíthetőek. Ezen a tisztán az oktatásnak alárendelt, kontrollált - és ily módon meglehetősen atipikus - játékon nőtt föl a magyar informatikusok első nemzedéke.

A számítógépes játék viszont lényegéből adódóan alapvetően szoftver. De mi is a játék? A két kiváló kibernetikus, Csákány Antal és Dr. Vajda Ferenc a következő - nem túl eredeti, de valóban igen fontos - elvárásokat kéri számon a játékon és azon belül a számítógépes környezetben zajló változatokon is:

"- a játék nem tartozik közvetlenül az életfenntartás folyamataihoz, de mégsem felesleges időtöltés; - a játék szabad cselekvés; a játékosok önállóan, kényszer nélkül vállalkoznak rá; a kötelező játék nem játék; - a játékok legnagyobb részénél a játékosok lelkesedése, megszállottsága a mozgató erő, így a játék többnyire érdek nélkülinek tűnik (ezt még a szerencsejátékok esetében is igaznak tartjuk: az igazi lóverseny-fogadó

nagyobb nyereség után sem hagyja abba a játékot); - a játék mindig zárt rendszer, vagyis elhatárolt, térbelileg és szabályait tekintve a valóságtól határozottan elkülönül. A játék szabályai áthághatatlanok, nincsenek 'kiskapuk'; - a játék nem nélkülözi a drámai elemeket: a konfliktushelyzet feszültséget teremt, a feszültség szabályozza a küzdelmet, a küzdelem végén megtaláljuk a feloldást, a sikerélményt.

(A vesztes is mindig talál a maga számára valamilyen magyarázatot.)"

Nos, ez az idézet épp olyan rendes és tiszta, mint a szellemi környezet, amelyben íródott. Tetszőlegesen kiegészíthetjük még (például a rekreáció, a szórakozás, a mintaelsajátítás ok-okozati viszonyaival), s bátran zárójelbe tehetjük szimbolikusan is a szerencsejátékosra vonatkozó bájosan naiv megjegyzést.

Összességében figyelemre méltó, hogy ezen sablon birtokában a szerzőpáros egy sor - főleg logikai és főleg hagyományos, azaz játék-elődökkel, gyökerekkel rendelkező - számítógépes játékot ismerttet, némelyiknek, így a Star Trek nevű stratégiai programnak a kultuszát is elemzi vagy legalábbis említi (bajnokságok, divat, stb.).

S mi indítja a szerzők szerint "a számítógéphez hozzáférő szakembereket (sic! kiemelés K.G.) számítógépes-játékok írására? Elsősorban talán az a kísérteties gondolat: hátha sikerül valami olyat létrehozni, ami - legalább egy szűk területen, a játék szabályrendszerében - okosabb, mint az alkotója. (...) A mesterséges intelligencia kutatás célja az, hogy megismerjék és leírják az emberi intelligencia azon jellegzetes vonásait, amelyeket a számítógép utánozni képes. Ezek közé tartoznak a játékok is, itt tehát a játékok elemzése során az emberi gondolkodás alapvető jellegzetességeit kutatják."

Íme, egy alkalmazott elmélet a számítógépes játékról és ez a megközelítés végül két módszerre vezet vissza az összes játékprogramot:

diszkrét helyzetek sorozatából álló játékokra (mint amilyenek a kalandjátékok) és differenciáljátékokra (mint amilyenek az ügyességi, reflexjátékok).

Magam is azon a véleményen vagyok, hogy a mai játékok is e két kristálytiszta matematikai mechanizmus különböző arányú és elrendezésű keverékei. S hiába a háttorzongatóan élethű (és így harci szituációkat modellező vagy bizarr környezetben játszódó programok esetén bizonyos életkor alatt vagy bizonyos lelki sérülékenység esetén veszélyes) képi világ, a filmről vagy a jövőben akár hologramról digitalizált látvány, az emberi beszédet szintetizáló hang: a program, ha meztelen, szinte teljesen ugyanolyan, mint volt a könyv írásakor, a nyolcvanas évek elején.

Ted Friedman, aki nálamnál bátrabb módon már a kilencvenes évek változásait is forradalomnak nevezi, a számítógépes játékot elsősorban az irodalomelmélet nézőpontjából és terminológiai vértetével vizsgálva, egy játéktörténet, egy kialakulás-történet vázlatát is adja, amelyet én csak néhány aprósággal egészítenék ki, saját szerény hozzászólásaimat zárójelbe is utasítva.

Első lépés: 1962-ban készül el az első játék, a Spacewar. Ezt az asteroids-típusú játékot unatkozó programozók készítik egy 120 ezer dolláros PDP-1 gépen. (A zenélő számítógép is az unalom terméke: a futásellenőrzés könnyítésére és az éjszakai műszakban pihegő kollégák szórakoztatására találta ki az első zenei programot Kovács Győző és néhány munkatársa az M-3 elektroncsöves számítógépen).

A Spacewar már példázta, hogy (miként Csákány és Vajda is utal rá) a játék az ember-gép interakció legjobb modellje, másrészt pedig a gép képességeinek egyik, ha nem a legjobb demonstrálója a nagyközönség előtt. Második lépésként a sakkprogramok és a szöveges kalandjátékok, így az Adventure jelennek meg (a hatvanas években, amikor a nagyszámítógépekkel egyre több fiatal programozó is kapcsolatba léphetett terminálokön keresztül; a jövő számítástechnikáját is sok hatalmas számítóközpont és milliónyi semmi-tudású terminál kapcsolataként képzelték el).

1970-ben jelent meg az első, bár kissé még passzív szimulációs játék, a Life.

A hetvenes évektől jelennek meg a videojátékok, egy külön dimenziója a játéknak és a célszámítógépek alkalmazásának. (Az első Ralph H. Baer Magnavox Odyssey készüléke, még a fekete-fehér televíziókhoz szánt színes előlapokkal. A leghíresebb pedig az Atari cég Pongja.) A hetvenes évektől jelenik meg a személyi számítógép (és annak különböző kategóriái, az eldobhatótól, a zsebgépen át a professzionálisig).

S a Friedman-cikk keletkezésekor, 1992-ben évente négyezer hivatalos és több tízezer szabad terjesztésű (public domain) játék került forgalomba.

A számítógépes játéknak saját szubkultúrája, kritikus gárdája, ikonjai, szakzsargonja, folyóiratai jöttek létre. A játékipar Új Hollywood lett, s miként egy másik szerző, Beregi Tamás megjegyzi, míg korábban, így a Star Trek esetén a filmipar adott ötletet és ösztönzést a játékiparnak, ma ez egyre inkább fordítva történik (s tegyük hozzá: egyelőre a film minőségének, gondolatgazdagságának rovására).

Kialakultak a főbb játéktípusok is, így az akció/arcade, a kaland, a szerepjáték, a szimuláció, a sport, illetve a hagyományos táblás játék.

A szimulációs játékok egy speciális formája, mint a Sim City, egyben különös ideológiai felépítményt is alkot. S megjelenik a számítógépes játék pszichológiai kritikája is: hisz míg a játékok cserélgetése és a róluk való diskurzus tagadhatatlanul új, fiatalos és érdekes közösségeket teremt, maga a játék elidegenít, amennyiben mintaként a Magányos Hős képét adja át a játékosának.

A kilencvenes évek végén megjelenik az online játék, amely földrajzilag egymástól hihetetlenül messze élő embereket hoz egymáshoz közel, de rendszerint úgy, hogy leplezi a játékos igazi személyiségét. Az online szerepjátékokban, amelyek elődje az 1979-es Dungeon Master, a játékos bábuvá válik, s az egeret irányító kéz apró mozdulataival úgymond hatalmas tettekre lehet képes. Ez a felelősségérzetre vagy a brutalitás érzékelésére káros hatással is lehet.

Az ismert amerikai szociológus, Theodore Roszak szerint a hetvenes évek számítógépes mozgalma igazi ellenkultúra volt, a nagy elődök, így a beat nyomán. Kommunisztikus nemzedéki életérzés és egy nagy amerikai álom: a fiatalok helyzeti előnybe kerültek az idősebbekkel szemben, s a szerzői jogokra fittyet hányva cserélgetik programjaikat és közösen minden szabadidejüket egymásnak szentelik. Számítógépes játékkal játszani pompás szórakozás - de mellette a számítógépes játékot megtervezni, megírni egy programozási nyelven (tipikusan BASIC-ben): virtus. S közben néhányan közülük (a "játékosok" közül) - s ez itt már az amerikai álom - szemvillanás alatt milliómossá válnak.

KIRÁLYKISASSZONY KATONÁJA

Játékgyáros legjobbjai

NyitraszóFT
Játéklóhoz
katonája és a
királykisasszony

szöveges,rajzos

kalandjáték

játékgyáros 2004

A testorkatona kalandjai es
a kiralykisasszony puszi ja.
rajzos kalandjatek iskolasoknak

egy testor vagy, es egy finom
tortat ajandekozol a csinos
es szep kiranylany nak,
ezert sok-sok puszit kapsz ente.

Sikeres kuldetest kivanok !!!
a jatekgyaros....

1. Játékdoboz katonája - mint induló helyszín

A jatekdoboz katonaja



Itt ezek vannak: alma, cipo,
Nalad van : semmi
Iranyod: elore, jobb,
Parancs ? (felvesz, kidob, csere)

2. Kutyás öreganyó

Oreganyo kutya ja



Itt ezek vannak: edeny,
Nalad van : sepru,
Iranyod: elore, jobb, hatra, bal,
Parancs ? (felvesz, kidob, csere)

3. Óriási csizma-ház

Az oriaszi csizma-haz



Itt ezek vannak: torta, palca,
Nalad van : sepru,
Iranyod: előre, bal,
Parancs ? (felvesz, kidob, csere)

4. Három mackó konyhája

A három macko konyhaja



Itt ezek vannak: edeny, sepru,
Nalad van : semmi
Iranyod: előre, jobb, hatra, bal,
Parancs ? (felvesz, kidob, csere)

5. Jancsi és Juliska kútja

Jancsi es Juliska kutja



Itt ezek vannak: alma, ora,
Nalad van : semmi
Iranyod: előre, bal,
Parancs ? (felvesz, kidob, csere)

6. Égigérő paszuly



7. Három nővér kertje



8. Királykisasszony Palotája - mint végső helyszín



A különböző helyszíneken és az indulásnál - mindenhol két-két dolog legyen, ezeket lehet felvenni és magunkkal vinni, vagy letenni, vagy elcserélni más dolgokért.

Legyenek ezek:

1. pók
2. seprű
3. lámpa
4. edény
5. pálca
6. cipő
7. alma
8. tej
9. TORTA
10. csont
11. könyv
12. ceruza
13. óra
14. ásó

Természetesen szerepeltetni kell olyanokat, akik ezeket elfogadják pl. A seprű jól jön a Vasorrú Bábának, hogy repülni tudjon, a pók ijesztgetésre való, a csodalámpa és az Aladdinnak, a pálca a tündérnek, a cipő és a az edény az öregapónak, a ceruza és könyv a növéreknek, a csont a kutyás öreganyónak, az ásó a paszulyinak, az óra az óriásnak, az alma és a tej a kismackóknak stb.

A játék közben különböző irányokba lehet haladni, így:

1. előre
2. jobbra
3. balra
4. hátra

és szükséges még néhány parancs is : (minimálisra lerövidítve, hogy minél kevesebbet kelljen írni, persze elég az első betű is !!)

1. felvesz
2. kidob
3. csere

A játék különlegessége még, hogy iskolás gyermekeknek készült, és általuk is jól olvashatóan, lassan, különleges betűnként íródik ki minden, és ez a játékosokat is gyakoroltatja, írni és olvasni kell !!! így lehet a kéréseket, parancsokat is beírni. Az irányításnál és a parancsoknál elég az első betűt beírni, hogy azért egyszerűbb-gyorsabb legyen.

A játék során - természetesen véletlenszerűen, csalafinta kérdések következnek, és mint egy társasjátékban - nem lehet továbbmenni, vissza kell térni, helyesen kell válaszolni stb.



A játékban - a cél - természetesen meg kell keresni a tortát, (vagy meg kell jegyezni, hogy hol is volt elhelyezve ,ide később vissza lehet térni), és el kell jutni a Királykisasszony Palotájához, és ott át lehet átadni az ajándék tortát. Előfordul, hogy torta nélkül jutunk el a végső célig, de akkor a királykisasszony igen haragos lesz, és visszaparancsol a kiinduló helyre.



Ha sikerrel eljutottunk a Palotába és a torta is nálunk van, akkor sok-sok puszi lesz a jutalmunk és egy kis győzelmi zenét is hallgathatunk.

```

1 REM Kiralykisasszony Palotaja
NyitraiSOFT jatekgyaros 2004
1 GOSUB 570: GOTO 3
20 INK 2: BEEP .01,15: LET zz=USR 63121: RETURN
3 RANDOMIZE : POKE 23658,0: POKE 23651,250: POKE
23652,250
5 LET xp=65184: LET yp=65185: LET yl=65186: LET
xl=65187
6 RESTORE 9
7 FOR x=0 TO 7: READ a: POKE 65368+x,a: NEXT x
8 REM a-RETURN-nyil karakter
9 DATA 28,28,28,28,127,62,28,8
10 LET l=1: DIM g(9): DIM h(2): DIM o(9): DIM
k(7,2): DIM f(9)
11 POKE 23609,75
14 LET g(1)=9
15 DIM c(8): FOR x=1 TO 8: LET c(x)=0: NEXT x:
LET g(9)=10
16 IF INT (RND*2)+1=2 THEN LET x=INT (RND*8)+1:
LET g(3)=x: LET c(x)=1: LET g(2)=9: GOTO 20
17 LET x=INT (RND*8)+1: IF c(x)=1 THEN GOTO 17
18 LET g(2)=x: LET c(x)=1
19 LET g(3)=9
20 LET x=INT (RND*8)+1: IF c(x)=1 THEN GOTO 20
21 LET c(x)=1: LET g(4)=x
22 IF INT (RND*2)+1=2 THEN GOTO 30
23 LET x=INT (RND*8)+1: IF c(x)=1 THEN GOTO 23
24 LET c(x)=1: LET g(5)=x: LET g(6)=9
25 GOTO 40
30 LET x=INT (RND*8)+1: IF c(x)=1 THEN GOTO 30
31 LET c(x)=1: LET g(6)=x: LET g(5)=9
40 LET x=INT (RND*8)+1: IF c(x)=1 THEN GOTO 40
50 IF INT (RND*2)+1=2 THEN GOTO 60
51 LET g(7)=x: LET c(x)=1: LET g(8)=9
52 GOTO 71
60 LET g(8)=x: LET c(x)=1: LET g(7)=9
71 LET g(9)=INT (RND*8)+1: IF c(g(9))=1 THEN GOTO
71
72 DIM c(14): IF g(3)=9 OR g(4)=9 THEN GOTO 76
73 LET x=INT (RND*3)+1
74 IF g(3)=9 THEN LET k(x,1)=g(4): LET c(g(4))=1:
GOTO 76
75 LET k(x,1)=g(3): LET c(g(3))=1
76 LET x=INT (RND*4)+1
77 IF g(5)=9 THEN LET k(x,2)=g(6): LET c(g(6))=1:
GOTO 80
78 LET k(x,2)=g(5): LET c(g(5))=1
80 FOR x=1 TO 7
81 FOR z=1 TO 2
82 LET y=INT (RND*14)+1: IF k(x,z)>0 THEN GOTO 88
83 IF c(y)=1 THEN GOTO 82
87 LET k(x,z)=y: LET c(y)=1
88 NEXT z
89 NEXT x
90 LET f(1)=1: LET f(9)=9
91 FOR u=2 TO 7
92 LET rnd=INT (RND*7)+1: IF rnd=1 THEN GOTO 92
93 IF f(u)>0 THEN GOTO 92
94 FOR x=2 TO 7
95 IF f(x)=rnd THEN GOTO 92
96 NEXT x
97 LET f(u)=rnd
98 NEXT u
99 REM a felveendo targyak 14 db 2
hosszon
100 DIM p(14,2)
105 RESTORE 125: FOR x=1 TO 14: FOR y=1 TO 2: READ
p1
106 LET p(x,y)=p1: NEXT y: NEXT x
125 DATA
233,84,233,112,233,160,233,208,234,0,234,48,234,96,23
4,144,234,192,234,240,235,32,235,80,235,128,235,176
189 REM ketoldalal zold fa
190 BORDER 7: PAPER 7: INK 0: CLS
191 POKE 23675,216: POKE 23676,232
192 PRINT AT 3,0; INK 4;"

```

```

193 PRINT AT 4,0; INK 4;"0000"
194 FOR x=5 TO 7: PRINT AT x,0; INK 4;"0000"
0000": NEXT x
195 PRINT AT 6,0; INK 4;"0000"
196 PRINT AT 9,0; INK 0;"0000"; INK 4;"6"
@"; INK 0;"00"
00"
197 FOR x=10 TO 11: PRINT AT x,0; INK 0;" 00"
00"
198 NEXT x
199 PRINT AT 12,0;" 00"
211 POKE xp,6: POKE yp,1: POKE xl,20: POKE yl,104:
POKE 64581,56: LET zz=USR 64560: POKE xl,32: POKE
yl,72: POKE xp,0: POKE yp,103: LET zz=USR 64560: LET
t$="": PRINT AT 21,0;:
GOSUB 2
212 IF l=6 AND h(1)=9 THEN GOTO 9950
213 IF l=8 AND h(2)=9 THEN GOTO 9950
218 IF l=8 THEN GOTO 9900
220 GOSUB 8000+(f(l)*100)
235 GOSUB 7600
237 GOTO 7010
255 DIM e(4): FOR x=1 TO 4: LET e(x)=0: NEXT x
260 GOSUB 7500+(l*10)
265 REM a jatek iranyitasa
270 PRINT AT 17,0;: LET T$="Irandod:"; GOSUB 2
280 LET T=8: IF e(1)=1 THEN PRINT AT 17,T;: LET
T$="elore,"; GOSUB 2: LET T=T+6
290 IF e(2)=1 THEN PRINT AT 17,T;: LET T$="jobb,";
GOSUB 2: LET T=T+5
300 IF e(3)=1 THEN PRINT AT 17,T;: LET
T$="hatra,"; GOSUB 2: LET T=T+6
310 IF e(4)=1 THEN PRINT AT 17,T;: LET T$="bal,";
LET t=t+5: GOSUB 2
320 REM a parancsok
330 PRINT AT 19,0;: LET T$="Parancs ?
(felvesz,kidob,cserere)": GOSUB 2: GOSUB 510
335 LET a$=U$
340 IF a$="kidob" OR a$="k" THEN GOSUB 7605: GOTO
7100
350 IF a$="cserere" OR a$="c" THEN GOSUB 7605: GOTO
7200
360 IF a$="felvesz" OR a$="f" THEN GOSUB 7605:
GOTO 7300
370 IF a$="e" OR a$="elore" OR a$="j" OR a$="jobb"
OR a$="h" OR a$="hatra" OR a$="b" OR a$="bal" THEN
GOTO 390
375 IF LEN a$>7 THEN GOTO 211
376 IF a$=" e" OR a$=" j" OR a$=" h" OR a$=" b"
THEN GOTO 390
380 GOTO 330
401 IF a$(1 TO 1)="e" THEN LET p=1
402 IF a$(1 TO 1)="j" THEN LET p=2
403 IF a$(1 TO 1)="h" THEN LET p=3
404 IF a$(1 TO 1)="b" THEN LET p=4
405 GOTO 9200+(l*10)
407 IF b=0 THEN GOSUB 7600: PRINT AT 15,0;: LET
t$="Ez itt nem a kizarat.": GOSUB 2: GOSUB 5000: GOTO
235
409 IF g(b)=9 THEN LET l=n: GOTO 211
410 IF o(b)=1 THEN GOSUB 7600: GOSUB
9700+(10*g(b)): PAUSE 100: LET l=n: GOTO 211
415 GOSUB 7600: GOSUB 9800+(g(b)*10)
418 GOSUB 510: LET b$=U$
419 IF b$="igen" OR b$="i" THEN GOTO 450
420 IF b$="nem" OR b$="n" THEN GOTO 230
431 GOTO 418
450 IF h(1)=g(b) THEN LET h(1)=0: LET l=n: LET
o(b)=1: GOTO 211
451 IF h(2)=g(b) THEN LET h(2)=0: LET l=n: LET
o(b)=1: GOTO 211
452 GOSUB 7600
453 REM a talalos kerdesre rossz
valasz
454 PRINT AT 15,0;: LET t$="A valaszod rossz,ezer
viszsa": GOSUB 2: PRINT AT 17,0;: LET t$="kell
lepned,nyomd az ENTER-t.": GOSUB 2: GOSUB 5000: GOTO
230

```

```

510 LET U#=""
515 LET a#=""
516 LET oszlop=-1
517 IF oszlop=-1 THEN PRINT AT 21,0;: LET t#="."
GOSUB 2: GOTO 525
520 PRINT AT 21,oszlop;: LET t#=a#+".": GOSUB 2
525 LET a#="INKEY#
526 IF a#="" THEN GOTO 525
527 BEEP .05,30
530 LET U=.05,a#
535 IF U=12 OR U=48 THEN GOTO 510
540 IF U=13 THEN RETURN
550 IF U<97 THEN GOTO 525
560 IF U>122 THEN GOTO 525
561 IF LEN U#>20 THEN GOTO 510
565 LET U#=U#+a#: LET oszlop=oszlop+1: GOTO 520
566 REM jatek bevezeto
570 PRINT AT 0,2;: LET t#="A testorkatona
kalandjai es": GOSUB 2: PRINT AT 2,4;: LET t#="a
kiralykisasszony puszija.": GOSUB 2: PRINT AT 4,0;
PAPER 5; INK 1;: LET t#="rajzos kalandjatek
iskolasoknak": GOSUB 2: PRINT AT 8,0;: LET t#="egy
testor vagu,es egy finom": GOSUB 2: PRINT AT 10,0;:
LET t#="tortat ajandekozol a csinos": GOSUB 2: PRINT
AT 12,2;: LET t#="es szep kiranylanynak.": GOSUB 2:
PRINT AT 14,0;: LET t#="ezert sok-sok puszt kapsz
erte.": GOSUB 2: PRINT INK 1;AT 18,0;: LET
t#="Sikeres kuldetest kivanok !!!": GOSUB 2
571 PRINT AT 20,7;: LET t#="a jatekgyaros....":
GOSUB 2
572 RETURN
4999 REM nyil karakter
5000 POKE 23675,88: POKE 23676,255: PRINT AT 21,31;
PAPER 6; FLASH 1;"@": IF INKEY#="" THEN GOTO 5000
5001 RETURN
6000 PRINT AT 0,0;"
": RETURN
7000 REM a hasznalati targyak
7010 LET q#="Itt ezek vannak: ": GOTO 7030
7011 LET q#=q#+ "pok,": RETURN
7012 LET q#=q#+ "sepru,": RETURN
7013 LET q#=q#+ "lampa,": RETURN
7014 LET q#=q#+ "edeny,": RETURN
7015 LET q#=q#+ "palca,": RETURN
7016 LET q#=q#+ "cipo,": RETURN
7017 LET q#=q#+ "alma,": RETURN
7018 LET q#=q#+ "tej,": RETURN
7019 LET q#=q#+ "torta,": RETURN
7020 LET q#=q#+ "csont,": RETURN
7021 LET q#=q#+ "konyv,": RETURN
7022 LET q#=q#+ "ceruza,": RETURN
7023 LET q#=q#+ "eso,": RETURN
7024 LET q#=q#+ "ora,": RETURN
7030 IF k(L,1)=0 AND k(L,2)=0 THEN LET q#=q#+ "
semmi": GOTO 7040
7032 IF k(L,1)>0 THEN GOSUB 7010+(k(L,1))
7033 IF k(L,2)>0 THEN GOSUB 7010+(k(L,2))
7040 PRINT AT 13,0;: LET t#=q#: GOSUB 2
7050 LET q#="Nalad van : "
7060 IF h(1)=0 AND h(2)=0 THEN LET q#=q#+ " semmi":
GOTO 7070
7062 IF h(1)>0 THEN GOSUB 7010+(h(1))
7064 IF h(2)>0 THEN GOSUB 7010+(h(2))
7070 PRINT AT 15,0;: LET t#=q#: GOSUB 2
7080 GOTO 240
7120 IF h(1)=0 AND h(2)=0 THEN GOSUB 7600: PRINT AT
15,0;: LET t#="Nalad semmi sincs.": GOSUB 2: GOSUB
5000: GOTO 235
7130 IF k(L,1) AND k(L,2)>0 THEN GOSUB 7600: PRINT
AT 15,0;: LET t#="Ezen a helyen nem ": GOSUB 2: PRINT
AT 17,0;: LET t#="teheted le.Nyomd a RETURN-t": GOSUB
2: GOSUB 5000: GOTO 235
7140 PRINT AT 19,0;: LET t#="Mit is akarsz letenni
?": GOSUB 2: GOSUB 510: LET b#=U#
7141 LET sor=7160
7142 LET ugras=7153: GOTO 9010
7153 POKE 23658,0: GOTO 7135

```

```

7160 IF NOT h(1)=j AND NOT h(2)=j THEN GOSUB 7600:
PRINT AT 15,0;: LET t$="Nincs mit letenned. ": GOSUB
2: GOSUB 5000: GOTO 7135
7165 IF h(1)=j THEN LET h(1)=0
7170 IF h(2)=j THEN LET h(2)=0
7175 IF k(l,1)=0 THEN LET k(l,1)=j: GOTO 7180
7176 LET k(l,2)=j
7180 GOTO 235
7210 IF h(1)=0 AND h(2)=0 THEN GOSUB 7600: PRINT AT
15,0;: LET t$="Nem cserelhetsz el semmit.": GOSUB 2:
GOSUB 5000: GOTO 235
7215 IF k(l,1)=0 AND k(l,2)=0 THEN GOSUB 7600:
PRINT AT 15,0;: LET t$="Itt nincs mit elcserelned.":
GOSUB 2: GOSUB 5000: GOTO 235
7220 PRINT AT 19,0;: LET t$="Mit akarsz
letenni/eldobni?": GOSUB 2: GOSUB 510: LET b$=U$
7222 LET ugras=7234: LET sor=7235: GOTO 9010
7234 POKE 23656,0: GOTO 7220
7235 IF h(1)=j OR h(2)=j THEN GOTO 7260
7236 GOSUB 7600: PRINT AT 15,0;: LET t$="Ezt itt
nem teheted le.": GOSUB 2: GOSUB 5000: GOTO 230
7261 PRINT AT 19,0;: LET t$="Mit akarsz itt
felvenni?": GOSUB 2: GOSUB 510: LET b$=U$
7263 IF b$="pok" THEN LET d=1: GOTO 7280
7264 IF b$="sepru" THEN LET d=2: GOTO 7280
7265 IF b$="lampa" THEN LET d=3: GOTO 7280
7266 IF b$="edeny" THEN LET d=4: GOTO 7280
7267 IF b$="palcsa" THEN LET d=5: GOTO 7280
7268 IF b$="cipos" THEN LET d=6: GOTO 7280
7269 IF b$="alsza" THEN LET d=7: GOTO 7280
7270 IF b$="tej" THEN LET d=8: GOTO 7280
7271 IF b$="torta" THEN LET d=9: GOTO 7280
7272 IF b$="csont" THEN LET d=10: GOTO 7280
7273 IF b$="konyv" THEN LET d=11: GOTO 7280
7274 IF b$="ceruza" THEN LET d=12: GOTO 7280
7275 IF b$="ora" THEN LET d=14: GOTO 7280
7276 IF b$="aso" THEN LET d=13: GOTO 7280
7279 GOTO 7261
7280 IF k(l,1)=d OR k(l,2)=d THEN GOTO 7290
7285 GOSUB 7600: PRINT AT 15,0;: LET t$="Itt nincs
cseretargy.": GOSUB 2
7286 GOSUB 5000: GOSUB 7600: GOTO 230
7290 IF k(l,1)=d THEN LET k(l,1)=j
7291 IF k(l,2)=d THEN LET k(l,2)=j
7292 IF h(1)=j THEN LET h(1)=d
7293 IF h(2)=j THEN LET h(2)=d
7294 GOTO 235
7300 REM ha kettonel tobb          tárgy van
nalunk, hibauzenet
7320 IF h(1)>0 AND h(2)>0 THEN GOSUB 7600: PRINT AT
15,0;: LET t$="Van mar ket tárgy nálád!": GOSUB 2:
GOSUB 5000: GOTO 235
7330 IF k(l,1)=0 AND k(l,2)=0 THEN GOSUB 7600:
PRINT AT 15,0;: LET t$="Itt nincs mit felvenned. ":
GOSUB 2: GOSUB 5000: GOTO 235
7342 PRINT AT 19,0;: LET t$="Mit akarsz itt
felvenni?": GOSUB 2: GOSUB 510: LET b$=U$
7350 LET sor=7370: LET ugras=7362: GOTO 9010
7362 GOTO 7340
7370 IF k(l,1)=j OR k(l,2)=j THEN GOTO 7380
7372 GOSUB 7600: PRINT AT 15,0;: LET t$="Itt nincs
ilyen tárgy.": GOSUB 2: GOSUB 5000: GOSUB 7600: GOTO
235
7380 IF k(l,1)=j THEN LET k(l,1)=0: GOTO 7385
7382 LET k(l,2)=0
7385 IF h(1)=0 THEN LET h(1)=j: GOTO 235
7386 LET h(2)=j: GOTO 235
7510 LET e(1)=1: LET e(2)=1: RETURN
7520 LET e(1)=1: LET e(4)=1: RETURN
7530 LET e(2)=1: LET e(3)=1: RETURN
7540 LET e(1)=1: LET e(2)=1: LET e(3)=1: LET
e(4)=1: RETURN
7550 LET e(1)=1: LET e(4)=1: RETURN
7560 LET e(3)=1: LET e(4)=1: RETURN
7570 LET e(2)=1: LET e(3)=1: LET e(4)=1: RETURN
7580 LET e(2)=1: RETURN

```



```

000010 PRINT AT 10,6; INK 1; PAPER 7;"00"; PAPER
INK 0;"01"; INK 0;"0"; INK 0;"0"; INK 0;"0"; INK 0;"0"; INK 0;"0";
INK 0;"0"; INK 0;"0"; INK 0;"0"; INK 0;"0"; INK 0;"0"; INK 0;"0";
000020 PRINT AT 11,6; INK 1; PAPER 7;"00"; INK
000030 PRINT AT 12,6; INK 1; PAPER 7;"00"; INK
000040 RETURN
000050 POKE 23675,40:kutyaja
GOSUB 9300: PRINT AT 0,7;: LET t$="Oregonyo kutyaja":
GOSUB 9300: PRINT AT 0,6; INK 4;"0"; INK 0;"0"; INK 6;
PAPER 5;"0"; PAPER 2; INK
000010 PRINT AT 3,6; INK 4;"0"; PAPER 2; INK 0;"
000020 INK 7;"0"; PAPER 7; INK 0;"0"; PAPER 5;"0";
000030 PRINT AT 4,6; INK 4;"0"; PAPER 2; INK 0;"0";
000040 INK 4;"0"; INK 7;"0"; PAPER 7; INK
000050 PRINT AT 5,6; INK 4;"0"; PAPER 0; INK 7;"0";
000060 INK 4;"0"; PAPER 6; INK 0;"00";
000070 INK 7;"0"; PAPER 7; INK 0;"0"; PAPER 4;"0";
000080 PRINT AT 6,6; INK 4;"0"; PAPER 7; INK 0;"0";
000090 PRINT AT 7,6; INK 4;"0"; PAPER 7; INK 0;"0";
000100 PRINT AT 6,6; INK 0;"00"; PAPER 2; INK 1;"0";
000110 PRINT AT 7,6; INK 4;"0"; PAPER 7; INK 0;"0";
000120 PRINT AT 8,6; INK 4;"0"; PAPER 7; INK 2;"0";
000130 INK 1;"0"; PAPER 2;"0"; PAPER 1;"0";
000140 PRINT AT 1,6; INK 4;"0"; PAPER 7; INK 2;"
000150 INK 1;"00"; PAPER 5;"0"; INK 1;"00"; INK 4;"0"
000160 PRINT AT 10,6; PAPER 4;"0"; PAPER 7; INK 2;"
000170 PAPER 5;"0"; INK 2;"0"; INK 1;
000180 PRINT AT 11,6; PAPER 4; INK 7;"0"; PAPER 7;
INK 4;"000"; PAPER 4; INK 7;"0"; INK 1;"0"
000190 PRINT AT 12,6; PAPER 4; INK 0;"00";
000200 RETURN
000210 REM az
000220 POKE 23675,55: csizmahaz
GOSUB 9300: PRINT AT 0,5;: LET t$="az oriasi csizma-
han": GOSUB 9300: PRINT AT 2,6; PAPER 5; INK 2;"
000010 INK 4;"0";
000020 PRINT AT 3,6; PAPER 5; INK 3;"0"; INK 4;"0";
000030 INK 2;"000"; PAPER 5;"0"; INK 4;"0";
000040 PRINT AT 4,6; PAPER 5; INK 3;"0"; INK 4;"0";
000050 INK 3;"0"; INK 1;"000"; PAPER 4;"0";
000060 PRINT AT 5,6; PAPER 3; INK 4;"00"; PAPER
000070 INK 0;"0"; PAPER 4;"0"; PAPER 0; INK 1;"0"; PAPER
000080 INK 1;"00"; PAPER 4;"0"; PAPER 4; INK 7;"0"
000090 PRINT AT 6,6; PAPER 1;"0"; INK 0;"00"; PAPER 6;
000100 INK 3;"0"; PAPER 1;"0"; INK 2;"0"; PAPER 6;
000110 PRINT AT 7,6; PAPER 4; INK 1;"00"; INK 7;"0";
000120 INK 1;"00"; PAPER 4;"0"; PAPER 6; INK
000130 INK 2;"0"; INK 1;"0"; PAPER 4;"0"; PAPER 1; INK
000140 PRINT AT 8,6; PAPER 6; INK 1;"00"; PAPER 1;
000150 INK 1;"000"; PAPER 3; PAPER 4; INK 7;"0";
000160 PRINT AT 9,6; PAPER 4; INK 7;"0";
000170 INK 7;"000"; PAPER 4; INK 0;"0";
000180 PRINT AT 10,6; PAPER 4; INK 1;"00"; PAPER 1;
INK 7;"00"; INK 4;"000"; PAPER 4; INK 1;"00"; INK
000190 PRINT AT 11,6; PAPER 4;"000000"
000200 PRINT AT 12,6; PAPER 4;"000000"
000210 RETURN
000220 REM a harom macko konyhaja

```

```

004005 POKE 23675,223: POKE 23676,237: GOSUB 6000:
GOSUB 9300: PRINT AT 0,3: LET t$="A harom macko
KotyaJa": GOSUB 2
004100 PRINT AT 2,6: PAPER 4: INK 2;"0001a"; INK
0;"000a"
004200 PRINT AT 3,6: PAPER 1: INK 2;"a"; PAPER
5;"a"; PAPER 1; INK 5;"a"; PAPER 4: INK 2;"a"; PAPER
4: INK 0;"a"; PAPER 1; INK 0;"0"; PAPER 4: INK
0;"000a"
004300 PRINT AT 4,6: PAPER 6: INK 0;"0"; PAPER 4;
INK 2;"000a"; PAPER 6; INK 0;"0"; PAPER 4: INK 2;"000a";
004400 PRINT AT 5,6: PAPER 6: INK 0;"0"; PAPER 4;
INK 2;"000a"; PAPER 6: INK 0;"0"; PAPER 6; INK 0;"a";
004500 PRINT AT 6,6: PAPER 6; INK 0;"0"; INK 2;"a";
004600 PRINT AT 4,6: INK 6;"0"; PAPER 6; INK 0;"0000";
004700 PRINT AT 7,6: PAPER 6; INK 2;"0"; PAPER 1;
INK 0;"000a"
004800 PRINT AT 6,6: PAPER 5; INK 2;"0a"; INK
6;"a"; PAPER 6; INK 0;"0000a"; INK 5;"a"; INK
6;"a"; PAPER 5; INK 2;"0"; INK 6;"a";
004900 PRINT AT 9,6: PAPER 5; INK 5;"0";
005000 PRINT AT 10,6: PAPER 5; INK 2;"0"; PAPER 6;
INK 5;"0"; PAPER 5; INK 2;"0"; PAPER 6;
005100 PRINT AT 11,6: PAPER 5; INK 2;"0"; INK
0;"000000"
005200 PRINT AT 12,6: PAPER 5; INK 0;"0000000"
005300 RETURN
005400 REM Jancsi es Juliska kutja
005500 POKE 23675,135: POKE 23676,238: GOSUB 6000:
GOSUB 9300: PRINT AT 0,4: LET t$="Jancsi es Juliska
Kutja": GOSUB 2
005600 PRINT AT 2,6: PAPER 5; INK 3;"0"; INK 2;"0a";
PAPER 2; INK 1;"a"; PAPER 5;"00"; INK 2;"0"; PAPER 2;
INK 4;"0"
005700 PRINT AT 3,6: PAPER 5; INK 3;"0a"; INK
2;"a"; PAPER 2; INK 1;"0a"; PAPER 5;"0a"; PAPER 7;
INK 4;"0a"
005800 PRINT AT 4,6: PAPER 5; INK 3;"0"; INK 0;"a";
INK 0;"0"; INK 6;"0"; INK 0;"0"; INK
005900 PRINT AT 5,6: PAPER 5; INK 3;"0"; INK
3;"0a"; INK 3;"0"; PAPER 3; INK 4;"0a"
006000 PRINT AT 6,6: INK 3;"0a"; PAPER 3; INK 4;"00"
006100 PRINT AT 7,6: INK 3;"0"; PAPER 3; INK 6; INK
006200 PRINT AT 8,6: PAPER 3; INK 4;"0a"; PAPER
3; INK 0;"00"
006300 PRINT AT 9,6: PAPER 3; INK 4;"000a"; PAPER
3; INK 0;"0"
006400 PRINT AT 10,6: PAPER 3; INK 4;"a"; PAPER 4;
INK 1;"0000a"; PAPER 6; INK 4;"0a"; INK 0;"0"; INK
1;"a"
006500 PRINT AT 11,6: PAPER 4; INK 1;"a"; PAPER 7;
INK 1;"0a"; PAPER 4; INK 7;"0000a"; PAPER 6; INK
4;"a"; INK 0;"0a"; PAPER 1; INK 6;"a"
006600 PRINT AT 12,6: PAPER 7; INK 1;"0"; PAPER 4;
INK 7;"0000"; PAPER 2; INK 4;"a"; PAPER 6; INK 2;"a";
INK 0;"0"
006700 RETURN
006800 REM az egig ero paszuly
006900 POKE 23675,47: POKE 23676,239: GOSUB 6000:
GOSUB 9300: PRINT AT 0,6: LET t$="Az egig ero
Paszuly": GOSUB 2
007000 PRINT AT 2,6: PAPER 1; INK 7;"000a"; INK
4;"0a"; INK 2;"a"; INK 4;"0a"; PAPER 4; INK 1;"a";
PAPER 1; INK 4;"a"; INK 2;"0"; INK 4;"a"
007100 PRINT AT 3,6: PAPER 1; INK 7;"0"; INK
0;"0a"; INK 7;"a"; INK 0;"0a"; INK 4;"0"; PAPER 4;
INK 1;"a"; PAPER 1; INK 4;"00"
007200 PRINT AT 4,6: PAPER 1; INK 2;"0000a"; INK
7;"a"; INK 4;"0"; PAPER 4; INK 1;"a"; PAPER 1; INK
4;"a"
007300 PRINT AT 5,6: PAPER 1; INK 4;"0"; PAPER 7;
INK 0;"00"; PAPER 1; INK 4;"0a"; PAPER 4; INK 1;"a";
PAPER 1; INK 4;"0"; INK 2;"a"; INK 4;"a"

```



```

000011 IF p=2 THEN LET b=1: LET n=2: GOTO 406
000012 LET b=0: GOTO 406
000013 IF p=4 THEN LET b=1: LET n=1: GOTO 406
000014 IF p=1 THEN LET b=3: LET n=4: GOTO 406
000015 LET b=0: GOTO 406
000016 IF p=0 THEN LET b=4: LET n=4: GOTO 406
000017 IF p=3 THEN LET b=2: LET n=1: GOTO 406
000018 LET b=0: GOTO 406
000019 IF p=1 THEN LET b=6: LET n=7: GOTO 406
000020 IF p=2 THEN LET b=5: LET n=5: GOTO 406
000021 IF p=3 THEN LET b=3: LET n=2: GOTO 406
000022 LET b=4: LET n=3: GOTO 406
000023 IF p=1 THEN LET b=7: LET n=6: GOTO 406
000024 IF p=4 THEN LET b=5: LET n=4: GOTO 406
000025 LET b=0: GOTO 406
000026 IF p=3 THEN LET b=7: LET n=5: GOTO 406
000027 IF p=4 THEN LET b=8: LET n=7: GOTO 406
000028 LET b=0: GOTO 406
000029 IF p=2 THEN LET b=8: LET n=6: GOTO 406
000030 IF p=3 THEN LET b=6: LET n=4: GOTO 406
000031 IF p=4 THEN LET b=9: LET n=8: GOTO 406
000032 LET b=0: GOTO 406
003000 PRINT AT 0,0: LET t$=" ": GOSUB 2: RETURN
004000 REM csalafinta mondatok
be csapos kerdesek
00710 PRINT AT 15,0: LET t$="Te kesobb erkeztel
ide,mint": GOSUB 2: PRINT AT 17,0: LET t$="az
ijeszto Vasorru Baba,": GOSUB 2: PRINT AT 19,0: LET
t$="es nem adtad oda a pokot.": GOSUB 2: GOSUB 5000:
RETURN
00720 PRINT AT 15,0: LET t$="Te megint elkeztel,es
nem": GOSUB 2: PRINT AT 17,0: LET t$="adta d a
boszorkanynak a seprut.": GOSUB 2: GOSUB 5000: RETURN
00730 PRINT AT 15,0: LET t$="Te keson erkeztel,nem
adtad": GOSUB 2: PRINT AT 17,0: LET t$="oda
Aladdinnak a lampat.": GOSUB 2: GOSUB 5000: RETURN
00740 PRINT AT 15,0: LET t$="Ismet keson jottel,nem
adtad": GOSUB 2: PRINT AT 17,0: LET t$="oda az
oregaponak az edenyt.": GOSUB 2: GOSUB 5000: RETURN
00750 PRINT AT 15,0: LET t$="Megint keson
erkeztel,es nem": GOSUB 2: PRINT AT 17,0: LET
t$="adtad a palcat a tundernek.": GOSUB 2: GOSUB
5000: RETURN
00760 PRINT AT 15,0: LET t$="Elfelejtetted felvenni
a": GOSUB 2: PRINT AT 17,0: LET t$="ket cipot,amikor
atjottel": GOSUB 2: PRINT AT 19,0: LET t$="a saros
reten.": GOSUB 2: GOSUB 5000: RETURN
00770 PRINT AT 15,0: LET t$="Mielott ide ertel
volna,": GOSUB 2: PRINT AT 17,0: LET t$="ehes
voltal,es megetted a": GOSUB 2: PRINT AT 19,0: LET
t$="nalad levo almat.": GOSUB 2: GOSUB 5000: RETURN
00780 PRINT AT 15,0: LET t$="Sokaig tartott,mig
ideertel,": GOSUB 2: LET t$="nagyon szomjas voltal,":
PRINT AT 17,0: GOSUB 2: PRINT AT 19,0: LET t$="es
megittad a tejet.": GOSUB 2: GOSUB 5000: RETURN
00810 PRINT AT 15,0: LET t$="A Vasorru Baba vart
itt teged,": GOSUB 2: LET t$="elhozta neki a
pokot,": PRINT AT 17,0: GOSUB 2: LET t$="melyet
nagyon vart?": PRINT AT 19,0: GOSUB 2: RETURN
00820 PRINT AT 15,0: LET t$="A Vasorru Baba vart
itt teged,": GOSUB 2: PRINT AT 17,0: LET t$="nalad
van a sepru,mellyel": GOSUB 2: LET t$="tovabb akart
innen repulni?": PRINT AT 19,0: GOSUB 2: RETURN
00830 PRINT AT 15,0: LET t$="Aladdin var itt az
uton.": GOSUB 2: LET t$="Elhozta neki a
csodalampat": PRINT AT 17,0: GOSUB 2: LET t$="a
varazslashoz?": PRINT AT 19,0: GOSUB 2: RETURN
00840 PRINT AT 15,0: LET t$="Az oregapo var itt
rad.": GOSUB 2: PRINT AT 17,0: LET t$="O ruhakat
tisztitana.": GOSUB 2: PRINT AT 19,0: LET
t$="Elhozta az edenyt neki?": GOSUB 2: RETURN
00850 PRINT AT 15,0: LET t$="A tunder elvesztette a
palcajat.": GOSUB 2: LET t$="Ha elhozta neki,akkor
lehet,": PRINT AT 17,0: GOSUB 2: PRINT AT 19,0: LET
t$="hogy sikerrel jarsz majd?": GOSUB 2: RETURN
00860 PRINT AT 15,0: LET t$="Itt a mezon nagyon
saros az": GOSUB 2: PRINT AT 17,0: LET

```

```

t$="Ut,Hasznalod a ket cipot,mikor": GOSUB 2: LET
t$="atkelsz majd a mezon?": PRINT AT 19,0;: GOSUB 2:
RETURN
99070 PRINT AT 15,0;: LET t$="Mar sokaig vagy
uton,es ehes": GOSUB 2: LET t$="vagy,megeeszod az
albat?": PRINT AT 17,0;: GOSUB 2: RETURN
99080 PRINT AT 15,0;: LET t$="Te mar nagyon szomjas
vaug,": GOSUB 2: LET t$="megiszod a tejet?": PRINT
AT 17,0;: GOSUB 2: RETURN
99090 LET v$="a": GOSUB 9960
99100 REM ha nincs nalunk a torta
99110 FOR x=10 TO -10 STEP -1: BEEP .02,x: NEXT x
99120 LET t$="Sikeresen elerted a Palotat.": PRINT
AT 14,0;: GOSUB 2: PRINT AT 16,0;: LET t$="De nincs
nalad a torta.": GOSUB 2
99130 LET t$="A kiralylany merges,azt mondja,":
PRINT AT 18,0;: GOSUB 2: LET t$="sipirc vissza a
jatekdobozba!!!": PRINT AT 20,0;: GOSUB 2
99140 LET l=1: GOSUB 5000: CLS : GOTO 191
99150 LET v$="a": GOSUB 9960
99160 REM sikerult a Palotat megtalalnunk,es
a torta is nalunk van
99170 POKE 23675,128: POKE 23676,240
99180 GOSUB 9300: PRINT AT 0,2;: LET t$="A
kiralykisasszony palotaja": GOSUB 2: PRINT AT 2,6;
99190 INK 7;"000"; PAPER 7; INK 2;"a"; PAPER 7; INK
5;"0000a"
99200 PRINT AT 3,6; PAPER 5; INK 7;"000"; INK
0;"0000"
99210 PRINT AT 4,6; PAPER 5; INK 7;"000000a"; INK
2;"0000"
99220 PRINT AT 5,6; PAPER 5; INK 7;"000000"; INK
6; PAPER 6; INK 2;"a"; INK 6; PAPER 5;"00"
99230 PRINT AT 6,6; PAPER 5; INK 7;"000"; PAPER
5;"000a"; PAPER 6; INK 1;"0a"; PAPER
0;"00000000"
99240 PRINT AT 7,6; PAPER 5; INK 7;"00a"; PAPER
5;"00"; PAPER 6;"a"; PAPER 5;"00"; PAPER
5;"00"; PAPER 7; INK 0;"0a"; PAPER 5;"a"
99250 PRINT AT 8,6; PAPER 6; INK 5;"00"; PAPER 7;
INK 6;"0"; INK 6;"a"; INK 5;"00"; PAPER 7;
INK 6;"0"; INK 6;"a"; PAPER 5;"a"; INK 0; PAPER
7;"00a"; PAPER 6; INK 5;"a"
99260 PRINT AT 9,6; PAPER 6; INK 7;"00a"; INK
6; PAPER 7;"00"; PAPER 6; INK 7;"a"; INK 0;"00";
INK 7;"00"; PAPER 6; INK 7;"a"; INK 0;"00";
INK 0;"a"
99270 PRINT AT 10,6; PAPER 6; INK 7;"000a"; PAPER
6;"000"; INK 2;"a"; PAPER 0;"a"; PAPER
0;"00"; PAPER 6;"a"
99280 PRINT AT 11,6; PAPER 4; INK 7;"000a"; PAPER
4;"000"; PAPER 2;"00"; INK 6;"a"; PAPER 7;
INK 0;"00"
99290 PRINT AT 12,6; PAPER 4; INK 2;"000000"; PAPER
6;"a"; PAPER 2; INK 7;"00"
99300 RETURN
99310 REM a sikeres jatek utani
jutalom
99320 PRINT AT 13,0;: LET t$="Minden akadalyt
legyoztetel!": GOSUB 2
99330 PRINT AT 15,0;: LET t$="A kiralykisasszony
boldog,": GOSUB 2
99340 PRINT AT 17,0;: LET t$="hogy megajandekoztad a
tortaval!": GOSUB 2
99350 PRINT AT 19,0;: LET t$="Sok-sok puszit kapsz
ezert!!": GOSUB 2
99360 REM egy kis gyozelmi zene
99370 FOR x=1 TO 2: BEEP .11,7: BEEP .21,12: BEEP
.11,12: BEEP .11,11: BEEP .11,9: BEEP .11,11: BEEP
.21,12: BEEP .11,7: BEEP .21,7: NEXT x: BEEP .11,5:
BEEP .31,4: BEEP .351,2: BEEP .41,0
99380 PRINT AT 21,0;: LET t$="Megegyyszer
megprobaldod? i/n": GOSUB 2
99390 IF INKEY$="i" OR INKEY$="I" THEN RUN
99400 IF INKEY$="n" OR INKEY$="N" THEN GOTO 9998

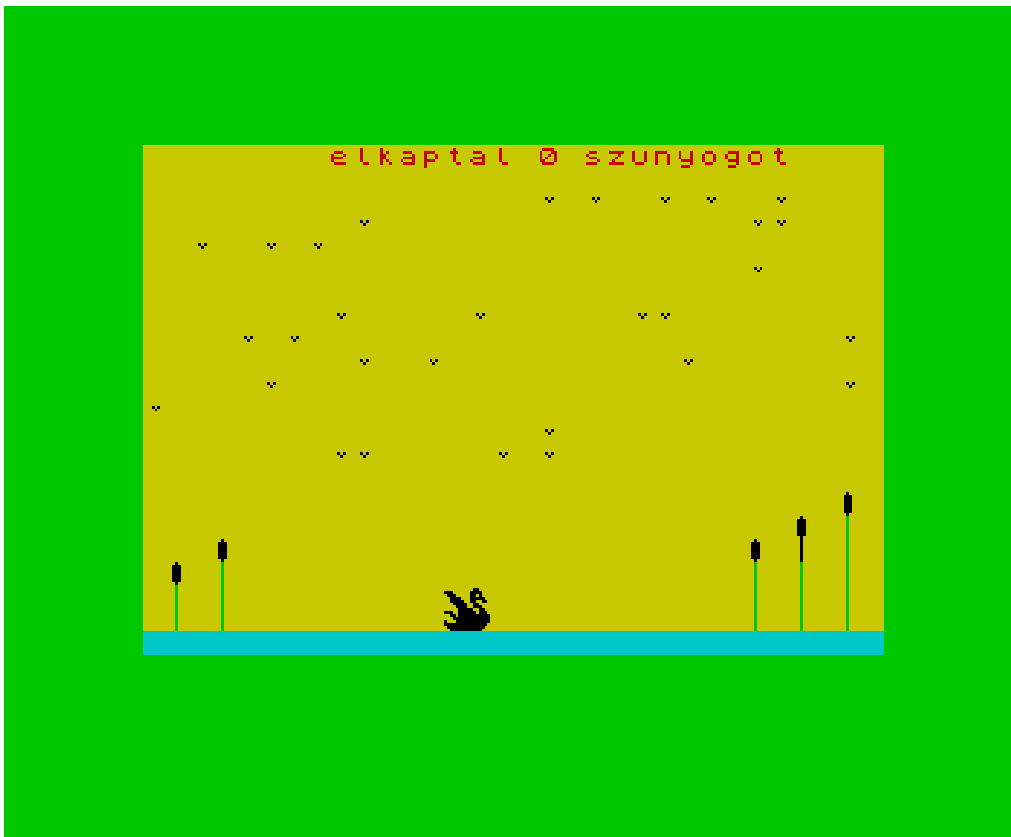
```

```

9991 GOTO 9989
9995 REM a spectrum elkoszon
9998 PRINT AT 21,0;: LET T$="Koszonom a finom
tortat!!": GOSUB 2: FOR n=1 TO 200: NEXT n: GOTO 9988
9999 SAVE "jatekdob" LINE 1: SAVE "kockakod"CODE
59608,5870

```

FEKETE HATTYÚ



- Dóri - unokám ,aki már 12 éves lesz májusban,vele leültünk megírni élete első BASIC programját.

Azt gondoltuk ki közösen,hogy folytatjuk a rút kiskacsa meséjét,,mikoris a rút kiskacsából egy szép,fekeete hattyú lett.A hattyú a nádasban gondtalanul él,és ott fogja ki a kis halakat,de életét megkeserítik amindenhol leskelődő,vérre éhes szúnyogok.Ezért szövetkezik a szintén ott fészkelő,gyors-szárnyú fecskével,és megbeszélik,hogy a hattyú ide-oda úszkál a nádas sekény vizén, és felveri az ottlévő,éppen a fecske elől elrejtőző szúnyogokat.Így azok felreppennek, és a fecskének már könnyű dolga van,szépen összefogdossa őket.Így mindenki jóljár,a szúnyogok nem csipkedik a hattyút, és a fecske is jól megebédel belőlük.

A programozás menete:

1./ Először mindig megtervezzük a "játékosokat" - a hattyút, a repülő fecskét, a nádat, és a szúnyogokat.

Erre legjobb a kockás papír, 8*8-as kockában elkészítjük a fentieket.

2./ Főcím, majd a magyarázó szöveg elkészítése, a program irányítása, 8000 sorban a nádas "felépítése"

3./ az alapértékek beállítása, szúnyog=0, szárny=0, hattyú és a fecske alappozíciója: d=19, a=4, m=3, n=28

A fenti értékek határozzák meg az ide-oda úszkáló hattyú és a fecske kezdő pozícióit

4./ a fecske repülése - irányítás beállítása 5=bal, 8=jobb, 6=le, 7=fel

5./ a 7000-es sor a szúnyogok véletlenszerű elhelyezkedése

6./ a fecske elfogja a szúnyogot 100 -as sorban, hangjelzés adása

7./ a sikeres elfogás kiírása

8./ ha sikerült mind a 30 szúnyogot elfogni, a játéknak vége

9./ Új játék indítás

```

1 REM Fekete hattyu 16-48-128K-s Spectrumra,basic
gyakorlat kezdőknek
2 REM készítette : Szabo Dora 9 éves kislány - a
jatekgyaros - segitsegevel
10 PRINT "Fekete hattyu": PAUSE 50
20 PRINT "A fekete hattyu szövetkezett a
gyorsroptü fecskevel,hogy a nadast megtisztitsak a
kinzo csipesü szunyogoktol.Iranyitas a
kurzor gombokkal.Nyomd az ENTER-t": PAUSE 0
25 REM az UDG-k beolvasasa
30 GOSUB 90000
35 REM a nadas felepitese
40 GOSUB 80000
45 LET szunyog=0: LET szarny=0
50 LET d=19: LET a=4: LET m=3: LET n=28
51 REM a szunyogok
55 GOSUB 70000
60 PRINT AT d,a;" AA";AT d+1,a;" AA"
75 PRINT AT m,n;" "
79 REM a fecske iranyitasa
80 LET m=m+(INKEY$="6" AND m<14)-(INKEY$="7" AND
m>1)
90 LET n=n+(INKEY$="8" AND n<31)-(INKEY$="5" AND
n>0)
95 IF INKEY$="5" OR INKEY$="6" OR INKEY$="7" OR
INKEY$="8" THEN LET szarny=szarny+1: BEEP .005,30
100 IF SCREEN$(m,n)="" THEN LET
szunyog=szunyog+1: BEEP .008,40
110 PRINT AT m,n;"B"
112 FOR i=1 TO 4: NEXT i
114 LET a=a+1: IF a>23 THEN PRINT AT d,a;" ";AT
d+1,a;" ": LET a=4
116 IF a<4 THEN LET a=4
120 PRINT INK 2;AT 0,8;"elkaptal ";szunyog ; "
szunyogot"
160 IF szunyog=30 THEN GOTO 1000
170 GOTO 60
1000 FOR f=0 TO 40: BEEP .008,f: NEXT f: CLS :
PRINT AT 11,0;"A fecskenek sikerult minden szunyogot
elfogni": PAUSE 150
1010 CLS : INPUT AT 11,2;"Ha jatszol ujra,nyomd az
ENTER-t"; LINE c#: GOTO 40
7010 FOR i=1 TO 30
7020 LET y=2+INT (RND*12): LET x=INT (RND*31)
7030 IF y=m AND x=n THEN GOTO 7020
7040 IF SCREEN$(y,x)="" THEN GOTO 7020
7050 PRINT AT y,x;"B"
7060 NEXT i: RETURN
7100 REM a nadas
8000 PAPER 6: BORDER 4: CLS
8010 FOR f=0 TO 31: PRINT INK 5;AT 21,f;"B": NEXT f
8110 PRINT AT 19,1; INK 4;"B";AT 19,3; INK 4;"B";AT
19,26; INK 4;"B";AT 19,28; INK 4;"B";AT 19,30; INK
4;"B"
8120 PRINT AT 20,1; INK 4;"B";AT 20,3; INK 4;"B";AT
20,26; INK 4;"B";AT 20,28; INK 4;"B";AT 20,30; INK
4;"B"
8130 PRINT AT 18,1;"B";AT 18,3; INK 4;"B";AT
18,26;"B";AT 18,28; INK 4;"B";AT 18,30; INK 4;"B"
8140 PRINT AT 17,3;"B";AT 17,26;"B";AT 17,28;"B";AT
17,30; INK 4;"B"
8150 PRINT AT 16,26;"B";AT 16,30; INK 4;"B"
8160 PRINT AT 15,30;"B"
8200 RETURN
8500 REM a hattyu,fecske , es a szunyog UDG-i
9000 FOR n=USR "a" TO USR "h"+7
9010 READ v: POKE n,v: NEXT n
9020 DATA 0,0,224,112,60,62,31,15
9030 DATA 0,40,120,104,124,102,48,56
9040 DATA 7,231,55,31,143,255,127,63
9050 DATA 204,238,255,255,255,254,252,248
9060 DATA 0,0,20,8,0,0,0,0
9065 DATA 16,16,16,16,16,16,16,16
9070 DATA 16,56,56,56,56,56,56,16
9080 DATA 1,5,106,220,60,2,1,0
9090 RETURN
9999 SAVE "hattyu":SAVE "hattyuUDG" CODE USR "a",8*8

```


RÓKA ÉS A LIBÁK (LUDAK ZX81)



```
1-REM NYITRAISOFT 2004
2-REM JATEKGYAROS                                1982-BEN ZX81-EN
2004-BEN SPECATULATOR                            EMULATORON
3-BORDER 1: PAPER 1: INK 7: CLS
5-LET A$="OUOUOUOUOUOUOUOUOUOUOUOUOUOUOUOUOU"
6-LET B$="⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪⓪"
10-PRINT AT 0,0;A$;AT 20,0;A$
11-PRINT AT 1,0;B$;AT 21,0;B$
14-REM A JÁTÉK MAGYARÁZATA
16-PRINT INK 6;AT 3,7;"ROKA ES A LIBAK..."
17-PRINT INK 6;AT 5,4;"EZ A JATEK EGY
KOZMONDAST";AT 6,3;"IDEZ,SOK LIBA ROKAT GYOZ."
18-PRINT INK 6;AT 8,4;"A FELADAT AZ,HOGY BE
KELL";AT 9,8;"KERITENI A ROKAT,";AT 10,4;"ES A
SAROKBA KELL ZARNI."
20-PRINT INK 6;AT 12,4;"IRANYITAS 5=BALRA,";AT
13,14;"8=JOBBRA";AT 14,4;"VALTANI A LIBAK";AT
15,4;"KOZOTT ""U""-VEL LEHET"
22-PRINT INK 6;AT 17,1;"ELOSZOR 4 LIBAD VAN,MAJD
SZÁM-";AT 18,1;"MUK CSOKKEN HAROMRA,KETTŐRE."
24-PRINT INK 2;AT 19,11;"KEZDHE TJUK ?"
25-PRINT INK 7;AT 11,25;" ou ";AT 12,25;" ⓪⓪ "
26-PRINT INK 7;AT 13,25;"ou"; INK 2;"Ug"; INK
7;"ou";AT 14,25;"⓪⓪"; INK 2;"?o"; INK 7;"⓪⓪"
27-PRINT INK 7;AT 15,25;" ou ";AT 16,25;" ⓪⓪ "
28-PAUSE 0
29-CLS
30-REM ROKA ⓪S A LIBAK
40-DIM d(10,10)
50-DIM x(4): DIM y(4)
60-DIM a(4): DIM b(4)
70-GO SUB 1370
80-GO SUB 940
90-LET h=INT (RND*3)+2: GO SUB 1110
100-GO SUB 650
110-GO SUB 140
120-GO TO 100
```

```

130-IF qj>6 THEN RETURN
140-PRINT AT 21,0;"A ROKA LEP"
150-IF d(qi+2,qj+2) AND d(qi+2,qj) AND d(qi,qj+2)
AND d(qi,qj) THEN GO TO 1420
160-LET m=1
170-GO SUB 410
180-IF qi+n<1 OR qi+n>8 THEN GO TO 210
190-IF d(qi+n+1,qj+m+1) THEN GO SUB 340
200-IF d(qi+n+2,qj+m+2)=1 AND d(qi+n,qj+m+2)=1 AND
qj<7 AND qj>1 THEN LET m=-m
210-IF d(qi+n+1,qj+m+1) THEN GO SUB 340
220-IF oi=qi AND oj=qj THEN LET m=-m: LET n=SGN
(RND-.5): GO TO 180
230-LET oi=qi: LET oj=qj
240-PRINT PAPER 7;AT qy,qx;" ";AT qy+1,qx;" "
250-LET qx=qx+n*20
260-LET qy=qy+m*20
270-PRINT PAPER 7; INK 2;AT qy,qx;"uy";AT
qy+1,qx;"?"
280-LET d(qi+n+1,qj+m+1)=2
290-LET d(qi+1,qj+1)=0
300-LET qi=qi+n
310-LET qj=qj+m
320-IF qj=0 THEN GO TO 1470
330-RETURN
340-LET n=-n
350-IF d(qi+n+1,qj+m+1)<>1 THEN RETURN
360-LET m=-m
370-IF qj<4 THEN LET n=1
380-IF d(qi+n+1,qj+m+1)<>1 THEN RETURN
390-LET n=-n
400-RETURN
410-LET n=(qi<5)-(qi>=5)
420-LET r=RND
430-IF r>.5 AND qi>3 THEN GO TO 470
440-IF qi>7 THEN GO TO 460
450-IF (d(qi+3,qj+3)=0 OR d(qi+2,qj+3)=0) AND
d(qi+2,qj+2)=0 THEN LET n=-1: RETURN
460-IF qi<4 OR r>.5 THEN RETURN
470-IF (d(qi-3,qj+3)=0 OR d(qi-2,qj+3)=0) AND d(qi-
2,qj+2)=0 THEN LET n=1: RETURN
480-IF r>.5 THEN GO TO 440
490-RETURN
500-LET a(hm)=a(hm)+m
510-LET b(hm)=b(hm)-1
520-IF d(a(hm)+1,b(hm)+1)<>0 THEN GO TO 560
530-LET d(a(hm)-m+1,b(hm)+2)=0
540-LET d(a(hm)+1,b(hm)+1)=1
550-GO TO 600
560-LET a(hm)=a(hm)-m
570-LET b(hm)=b(hm)+1
580-BEEP 1,-10
590-GO TO 650
600-PRINT PAPER 7;AT y(hm),x(hm); FLASH 0;" ";AT
y(hm)+1,x(hm);" "
610-LET y(hm)=y(hm)-2
620-LET x(hm)=x(hm)+m*2
630-PRINT PAPER 7; INK 1;AT y(hm),x(hm); FLASH
1;"ou";AT y(hm)+1,x(hm);"ou"
640-RETURN
650-PRINT AT 21,0;"L@P@SED? ": REM 3 HELY
660-BEEP .1,6
670-LET m#=INKEY#
680-IF m#="" THEN GO TO 670
690-IF m#="v" OR m#="V" THEN GO TO 730
700-IF m#="5" THEN LET m=-1: GO TO 500
710-IF m#="0" THEN LET m=+1: GO TO 500
720-GO SUB 770
730-PRINT AT 21,15;"VALTASZ? I"
740-LET a#=INKEY#
750-IF a#="D" OR a#="N" THEN GO TO 1470
760-GO TO 650
770-GO SUB 800
780-LET hm=hm+1
790-IF hm>h THEN LET hm=1
800-GO SUB 820
810-RETURN

```

```

0020-LET f=202500+x(hm)+y(hm)*32
0030-DOOK f,PEEK f+128
0040-DOOKMM f+1,PEEK (f+1)+128
0050-DOOKMMM f+30,PEEK (f+30)+128
0060-DOOKMMM f+33,PEEK (f+33)+128
0070-RETURN
0080-LET f=202500+x(hm)+y(hm)*32
0090-DOOK f,PEEK f-128
0100-DOOKMM f+1,PEEK (f+1)-128
0110-DOOKMMM f+30,PEEK (f+30)-128
0120-DOOKMMM f+33,PEEK (f+33)-128
0130-RETURN
0140-REM A TABLA
0145-PRINT AT 3,6;
0150-FOR i=1 TO 4
0160-FOR j=1 TO 6
0170-PRINT INK 3; PAPER 7;" 66";
0180-IF j/4=INT (j/4) THEN PRINT : PRINT TAB 8;
0190-NEXT j
1000-FOR j=1 TO 6
1010-PRINT INK 3; PAPER 7;"66 ";
1020-IF j/4=INT (j/4) THEN PRINT : PRINT TAB 8;
1030-NEXT j
1040-NEXT i
1050-INK 1: PLOT 63,23: DRAW 129,0: DRAW 0,129: DRAW
-129,0: DRAW 0,-129
1055-PLOT 61,21: DRAW 133,0: DRAW 0,133: DRAW -133,0:
DRAW 0,-133
1060-PLOT 59,19: DRAW 137,0: DRAW 0,137: DRAW -137,0:
DRAW 0,-137
1070-INK 1: FOR A=3 TO 11: PLOT 195+A,133: DRAW 0,-
120: DRAW -120,0: PLOT 76,20-A: DRAW 128,0: NEXT A
1090-PRINT INK 2;AT 10,26;"BAL =5";AT 11,26;"JOB=8";
INK 0;AT 13,26;"VALTAS"
1100-RETURN
1110-FOR i=1 TO h
1120-LET x=i*4+6
1130-PRINT INK 1; PAPER 7;AT 17,x;"ou";AT 18,x;"66"
1140-LET d(i*2+1,9)=1
1150-LET x(i)=x
1160-LET y(i)=17
1170-LET hm=1
1180-LET a(i)=i*2
1190-LET b(i)=8
1200-NEXT i
1210-GO SUB 820
1220-LET qi=5
1230-LET qj=1
1240-LET qx=qi*2+6
1250-LET qy=3
1260-PRINT PAPER 7; INK 2;AT qy,qx;"uy";AT
qy+1,qx;"?o"
1270-FOR i=1 TO 10
1280-LET d(i,1)=1
1290-LET d(1,i)=1
1300-LET d(10,i)=1
1310-LET d(i,10)=1
1320-NEXT i
1330-LET d(qi+1,qj+1)=2
1340-LET oi=0
1350-LET oj=0
1360-RETURN
1370-BORDER 4: INK 0
1380-PAPER 4
1390-CLS
1392-PRINT INK 2;AT 0,9;"ROKA "; INK 0;"ES A "; INK
1;"LIBAK";AT 1,8; INK 6;" BEKERITOS JATEK"
1400-BORDER 4
1410-RETURN
1420-PRINT INK 1;AT 21,0;"GYOZTEL,ELFOGTAD A ROKAT!!
"
1430-FOR n=0 TO 10
1440-BEEP .5,n
1450-NEXT n
1460-GO TO 1510
1470-PRINT INK 2;AT 21,0;"A ROKA KIJUTOTT,ES
GYOZOTT.... "

```

```

1480-FOR n=0 TO -10 STEP -1
1490-BEEP .5,n
1500-NEXT n
1510-INPUT "MEGEGYSZER ?";a$
1520-IF a#="i" OR a#="I" THEN RUN
1530-CLS
1540-PRINT AT 10,5;"SOK SIKERT KIVANOK, ";AT
12,9;"LEGKOZELEBB..."

```

ITT VAN AZ 1982-BEN KESZULT ZX81-ES OS-VALTOZAT,MELY NYOMTATASBAN MEGJELENT A BITLET-OTLET 1984-ES SZAMABAN

```

1 REM NYITRAI LASZLO 1982 LOGIKAI
5 REM DISZNO ES A LUDAK
JATEK
6 LET B#="00000000000000000000000000000000000000000000000000000"
8 LET C#=B$(1)+"
"+B$(1)
10 PRINT AT 0,0;B#;AT 21,0;B#
12 FOR N=1 TO 20
14 PRINT AT N,0;C#
15 NEXT N
16 PRINT AT 3,5;"DISZNO ES A LUDAK"
17 PRINT AT 5,2;"EZ A JATEK EGY KOZMONDAST";AT
6,1;" IDEZ,SOK LUD DISZNOT GYOZ."
18 PRINT AT 8,2;"A FELADAT AZ,HOGY BE KELL";AT
9,4;"KERITENI A DISZNOT, ";AT 10,2;"ES A SAROKBA KELL
ZARNI."
20 PRINT AT 12,2;"IRANYITAS A KURZORRAL, ";AT
13,5;"<-BALRA-JOBBRA->, ";AT 14,2;"UALTANI A LUDAK
KOZOTT ";AT 15,2;"A 7-ES NYILLAL LEHET."
22 PRINT AT 17,11;"KEZDHEJUK ?"
24 PAUSE 4E4
25 CLS
26 PRINT AT 1,5;"DISZNO ES A LUDAK"
29 PRINT AT 14,10;"▣?????????";AT
5,10;"▣?????????";
30 FOR I=1 TO 4
40 PRINT TAB 10;"? ? ? ? ?";TAB 10;"?? ? ? ? ?"
50 NEXT I
55 REM A DISZNO INDULO HELYZETE
KOZEPROL
60 LET P=PEEK 16396+256*PEEK 16397+33*6+12
70 LET H=P+3
80 POKE H,151
85 REM IRANYITAS
90 PRINT AT 8,24;"5 8";AT 9,20;"BALRA-JOBRA";AT
11,21;"7";TAB 23;"MASIK LUD";AT 14,21;"K-KILEPES"
95 REM A LUDAK A MEMORIABAN
100 DIM A(4)
110 FOR I=1 TO 4
120 LET A(I)=P+7*33+2*I-2
129 REM A LUD KODJA "L"
130 POKE A(I),177
140 NEXT I
150 LET L=H
160 FOR I=1 TO 4
165 REM ELLENORZES,HOGY URES KOCKA VAN-E
170 IF PEEK (A(I)-2)=128 AND PEEK (A(I)+32)=128 AND
PEEK (A(I)-34)=128 THEN LET L=A(I)-1
180 IF PEEK (A(I)+2)=128 AND PEEK (A(I)+34)=128 AND
PEEK (A(I)-32)=128 THEN LET L=A(I)+1
190 NEXT I
200 LET R=33+SGN (L-H-33*(INT ((L-P)/33)-INT ((H-
P)/33)))
210 IF R=33 THEN LET R=32+2*(PEEK (H+34)=128)
220 IF PEEK (H+R) <>128 THEN LET R=R-66
230 IF PEEK (H+R) <>128 THEN LET R=-R
240 IF PEEK (H+R) <>128 THEN LET R=R-66
250 IF PEEK (H+R) <>128 THEN GOTO 470
260 POKE H,128
270 LET H=H+R
275 REM A DISZNO KODJA "D"

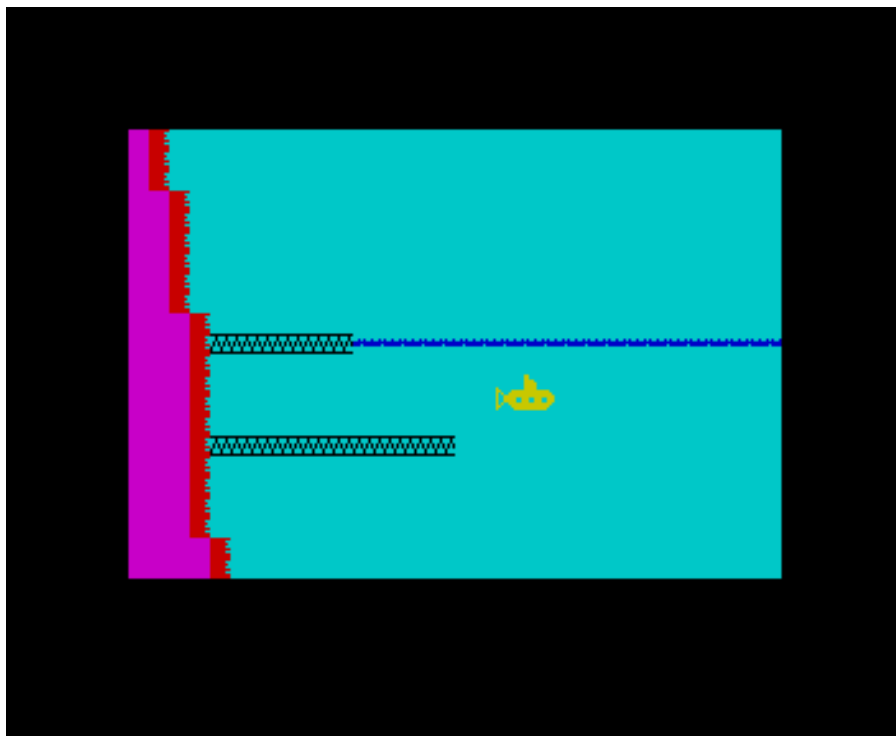
```

```

0000 POKE H,169
0005 REM A DISZNO KITORT,ES ELMENEKULT
0010 IF H>P+230 THEN GOTO 450
0015 FOR I=1 TO 4
0020 POKE A(I),364-PEEK A(I)
0025 LET I#=INKEY#
0030 REM LEPUNK BALRA ES JOBBRA
0035 IF I#="5" OR I#="8" THEN GOTO 390
0040 REM EZ BIZONY A FELADAS
0045 IF I#="K" THEN GOTO 450
0050 REM MASIK LUDDAL LEPUNK
0055 IF I#<>"7" THEN GOTO 310
0060 POKE A(I),177
0065 NEXT I
0070 GOTO 300
0075 REM CSAK EGYET LEPHETUNK
0080 LET R=-34+2*(I#="8")
0085 IF PEEK (A(I)+R)<>128 THEN GOTO 310
0090 POKE A(I),128
0095 LET A(I)=A(I)+R
0100 POKE A(I),CODE "L"
0105 REM KERGETODZUNK
0110 GOTO 150
0115 PRINT AT 16,5;"A DISZNO MEGSZOKOTT"
0120 PAUSE 4E4
0125 REM UJABB PROBA
0130 PRINT AT 16,5;" "
0135 GOTO 483
0140 REM HA SIKERULT MEGFOGNUNK, MI GYOZTUNK
0145 PRINT AT 16,3;"A LUDAK DISZNOT GYOZTEK"
0150 PAUSE 4E4
0155 PRINT
0160 PRINT " MEGEGYSZER ? I/ENTER-T"
0165 INPUT A#
0170 IF A#<>"I" THEN NEW
0175 GOTO 25
0180 SAVE "LUDAK"

```

SÁRGA TENGERALATTJÁRÓ - BEATLES -UTÁN SZABADON PROGRAMOZVA !!!



a SÁRGA II. tengeralattjárót irányítva meg kell keresni az elrejtett kincsesládákat, majd ki kell a polip fogságából szabadítani a SÁRGA I-et, irányítás a kurzorral, három életed van...

A program érdekessége, hogy képernyőként új-és új karakterkészletet használ, BASIC-ból így átlépve a bűvös 21 karakteres használatot, a programot különösképpen nem lassítja le, váltásoknál kell csak 1-2 másodpercet várni, míg az új képernyőkép megjelenik.

```

0-REM *****SARGA*****
*****
*****
** 2001 JANUAR **
*****
1-REM sarga tengeralattjaro          BEATLES rajzfilm
zeneje
2-DATA
0.75,11,0.25,12,2,14,0.75,14,0.25,11,0.75,9,0.25,11,3
,7,0.75,11,0.25,11,0.75,9,0.25,7,1,4,0.75,4,0.25,4,0.
75,11,0.25,11
3-DATA
3,9,0.25,11,0.75,12,2,14,0.75,14,0.25,11,0.25,9,0.75,
11,3,7,0.75,11,0.25,11,0.25,11,0.75,9,0.25,7,1,4,0.75,4,0.25,
4,0.75,11,0.25,11,3,9,0.75,11,0.25,12,2,14,0.75,14,0.
25,11,0.25,9,0.75,11,3,7,0.75,11,0.25,11,0.75,9,0.25,
7,1,4,0.75,4,0.25,4,0.5,11,0.75,11
4-DATA
3,9,0.75,11,0.25,12,2,14,0.75,14,0.25,11,0.75,9,0.25,

```

```

11,3,7,0.25,11,0.75,11,0.25,11,3,7,0.25,11,0.75,11,0.2
5,9,0,1,4,0.75,11,0.25,11,4,9
1,14,0,14,1,14,0.75,14,0.25,16,0.75,9,0.25,9,0.75,9,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1,14,0,14,1,14,0.75,14,0.25,16,0.75,9,0.25,9,0.75,9,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1,14,0,14,1,14,0.75,14,0.25,16,0.75,9,0.25,9,0.75,9,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1,14,0,14,1,14,0.75,14,0.25,16,0.75,9,0.25,9,0.75,9,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,7,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
10-IF FOR f=1 TO 150
11-READ a,b: BEEP a/3,b
12-NEXT f
15-REM NYITRAIL felirat es a magyarazat
tavoratban
30-RANDOMIZE USR 60000: CLS : GO SUB 9900: INK 1:
RANDOMIZE USR 60094: GO TO 75
31-LET a$="..TAVIRAT-A FEDELZETRE:
MOST @o EGY TENG U@UERA- LATTJARO ": RETURN
32-LET a$="KAPITANY VAGYA SARGA II.- VEL BEJAROD A
TENGEREK MELYET. ": RETURN
33-LET a$="4 ARANNYAL TELI LADIKOT KELL
MEGKERESNED EGY SOTETBARLANGBAN, MELYBE NEHEZ ":
RETURN
34-LET a$="LESZ BEHATOL-NI,MERT LAN- COS
HORGONYOKVEDIK A BE-JARATOT, AHOL A LADIKA VAN ":
RETURN
35-LET a$=" IRANYITAS: KEMPSTON-
JOYSTICK ": RETURN
36-LET a$=" VAGY : KURZOR NYILAK
(5/6/7/8) HASZNALATA. ": RETURN
37-LET a$="0 - TUZ CSAK EGYSZER, AZ ORIASPOLIP
ELLEN ": RETURN
38-LET a$="A LADIKAT3 KERESVE MINDEFELE TENGERI
SZORNYEKET FOGSZ LATNI. ": RETURN
39-LET a$="KI KELL SZABADITANOD A SARGA I.-ET A
POLIP KARJAIBOL..... SZERZO: NY.L.": RETURN
75-FOR s=1 TO 9: GO SUB ad: LET ad=ad+1
80-FOR d=1 TO 6: FOR f=1 TO 13: BEEP 0.02,30: PRINT
AT 9+d,16+f;a$(f): NEXT f: LET a#=a$(14 TO ): NEXT d
82-PAUSE 50: NEXT s
83-CLS : PRINT INK 2;AT 11,17;"KEMPSTON- ";AT
12,17;"JOYSTICK ";AT 13,17;" (I/N)"
84-IF INKEY#<>" THEN GO TO 84
85-IF INKEY#=" THEN GO TO 85
86-IF INKEY#="I" OR INKEY#="i" THEN LET joy=0
87-PAUSE 100
88-PAPER 5: INK 6: CLS
89-GO TO pos
90-STOP
100-IF NOT joy THEN GO TO 150
101-REM iranyitas Kempston es kurzor
105-IF INKEY#="0" THEN GO SUB 200
110-IF INKEY#="5" THEN GO SUB 300
120-IF INKEY#="6" THEN GO SUB 400
130-IF INKEY#="7" THEN GO SUB 500
140-RETURN
150-IF IN 31=1 THEN GO SUB 200
160-IF IN 31=2 THEN GO SUB 300
170-IF IN 31=4 THEN GO SUB 400
180-IF IN 31=8 THEN GO SUB 500
190-RETURN
2000-LET b=b+1: IF b=28 THEN GO TO 600
2005-BEEP .0005,46: LET li=0
2010-REM mozgatas es utkozések
2010-PRINT AT a,b-1; INK 6;" @o";AT a+1,b-1;" u@u"
2020-LET att=ATTR (a,b+3)
2021-IF att=111 THEN GO TO 900

```

```

0022-IF att<>42 AND att<>45 AND att<>46 THEN GO TO
0025-LET att=ATTR (a+1,b+3)
0026-IF att=111 THEN GO TO 900
0027-IF att<>42 AND att<>46 AND att<>45 THEN GO TO
0030-RETURN
0031-LET b=b-1: IF b=1 THEN GO TO 650
0032-BEEP .001,39: LET li=1
0033-PRINT AT a,b; INK 6;"[? ";AT a+1,b;"y[ "
0034-LET att=ATTR (a,b-1)
0035-IF att=111 THEN GO TO 900
0036-IF att<>42 AND att<>46 AND att<>45 THEN GO TO
0039-LET att=ATTR (a+1,b-1)
0040-IF att=111 THEN GO TO 900
0041-IF att<>46 AND att<>42 AND att<>45 THEN GO TO
0044-RETURN
0045-LET a=a+1: IF a=20 THEN GO TO 700
0046-BEEP .001,35
0047-IF li=1 THEN GO TO 415
0048-PRINT AT a,b; INK 6;"[o";AT a+1,b;"U@U";AT a-
1,b;" ": GO TO 420
0049-PRINT AT a,b; INK 6;"[?";AT a+1,b;"y[";AT a-
1,b;" "
0050-LET att=ATTR (a+2,b+1): IF att<>42 AND att<>46
AND att<>45 THEN GO TO 800
0051-RETURN
0052-LET a=a-1: IF a<=1 THEN GO TO 750
0053-BEEP .0005,49
0054-IF li=1 THEN GO TO 515
0055-PRINT AT a,b; INK 6;"[o";AT a+1,b;"U@U";AT
a+2,b;" ": GO TO 520
0056-PRINT AT a,b; INK 6;"[?";AT a+1,b;"y[";AT
a+2,b;" "
0057-LET att=ATTR (a-1,b+1): IF att<>42 AND att<>46
AND att<>45 THEN GO TO 800
0058-RETURN
0059-LET pos=pos+100
0060-LET b=2
0061-GO TO pos
0062-LET pos=pos-100
0063-LET b=27
0064-GO TO pos
0065-LET pos=pos+800
0066-LET a=1
0067-GO TO pos
0068-LET pos=pos-800
0069-LET a=19
0070-GO TO pos
0071-IF li=1 THEN GO TO 802
0072-FOR f=1 TO 30: PRINT AT a,b; INK 6;"[o";AT
a+1,b;"U@U";AT a,b;" ";AT a+1,b;" ": BEEP .005,f:
NEXT f: GO TO 810
0073-FOR f=1 TO 30: PRINT AT a,b; INK 6;"[?";AT
a+1,b;"y[";AT a,b;" ";AT a+1,b;" ": BEEP .005,f:
NEXT f:
0074-INK 0: FOR f=15 TO 0 STEP -1: PLOT b*8,(22-a)*8-
f: DRAW 24,0: NEXT F
0075-INK 6
0076-PAPER 5: INK 0: CLS
0077-REM utkozesekek utani utasitasok
0078-PRINT AT 6,10; INK 6;"[?";AT 7,10;"y["; INK
0;"3"; INK 7;"....."; INK 2;AT 10,8;"A
KINCSKERESEST";AT 12,8;"FOLYTATNOD KELL!"; INK 1;AT
14,8;"A MASIK";AT 16,8;"TENGERALATTJAROT,";AT 18,8;
INK 2; FLASH 1;"IS KERESD MEG!"
0079-PRINT INK 1;AT 20,8; FLASH 0;"FOLYTATOD ?"; INK
2;"(I/N)"
0080-IF INKEY#<>" " THEN GO TO 845
0081-IF INKEY#="" THEN GO TO 846
0082-IF INKEY#="i" OR INKEY#="I" THEN GO TO 890
0083-CLS: FLASH 1: PRINT INK 2;AT
10,8;"VISZONTLATASRA!": FLASH 0
0084-STOP
0085-CLS : GO SUB 9900: PAUSE 100: CLS : GO TO 83

```



```

900-PAPER 3: CLS : PRINT AT 6,4; INK 0;"MEGTALALTAD
D KINCSESSES-" ;AT 8,4;"LADAT,DE A SARGA I.-ET";AT
10,4;"MEG KI KELL";AT 12,4;"SZABADITANOD, ";i$;" MEG
D HAJOD!": PAUSE 350
9015-LET U(q)=0
9017-IF i$="nincs" THEN GO TO pos
9018-LET i=i+1
9019-INK 7
9021-REM i=a kincsesesladak szama
9025-IF i=2 OR i=3 OR i=4 THEN GO TO 950
9027-CLS : PRINT AT 4,4; INK 0;"A JAVITAST SIKERULT
";AT 6,4;"IDORE BEFEJEZNI, ";AT 8,4;"HAJOD UJ NEVE
SARGA II.": LET fire=1: GO TO 960
950-CLS : PRINT AT 6,4; INK 0;"SIKERULT MAR ";5-i;AT
7,4;"LADAT MEGTALALNI";AT 8,4;"DE JO LENNE MEG ";AT
9,4;"POLYTATNI A KUTATAST."
960-INK 6: FOR f=1 TO 20: BEEP f/20,f: NEXT f: GO TO
pos+100
1000-DATA
255,0,255,2,254,252,255,239,0,0,0,255,255,255,34,34,85,85,136,1
36,0,255,0,41,255,239,0,0,0
1001-RESTORE 1000: FOR f=USR "n" TO USR "p"+7: READ
M: POKE f,M: NEXT f: REM N=0 M=3 O=1 P=0
1002-PAPER 5: CLS : INK 3: FOR l=0 TO 21: PRINT AT
l,0;"i";AT l,1;"i" AND l>1;AT l,2;"i" AND l>8;AT
l,3;"i" AND l>19: NEXT l
1003-INK 2: FOR l=0 TO 21: PRINT AT l,1;"0" AND l<3
AND l>=0;AT l,2;"0" AND l<9 AND l>2;AT l,3;"0" AND
l<20 AND l>8;AT l,4;"0" AND l<=21 AND l>19: NEXT l
1004-GO SUB 6000: PRINT AT 10,4; INK 0; OVER
1;"11111111"; INK 1; OVER 1;"oooooooooooooooooooooooo";AT
15,4; INK 0; OVER 1;"11111111111111": IF iop=0 THEN
PRINT AT a,b; INK 6;"@b0";AT a+1,b;"U@U": LET iop=1
1006-LET att=46
1007-GO SUB 100: GO TO 1007
1100-DATA 0,0,41,255,239,0,0,0
1101-RESTORE 1100: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: REM N=0
1102-PAPER 5: CLS : PRINT AT 10,0; INK 1; OVER
1;"oooooooooooooooooooooooooooooooooooooooo"; GO SUB 9100: GO
SUB 6000
1104-GO SUB 100: GO TO 1104
1200-DATA 0,0,41,255,239,0,0,0
1201-RESTORE 1200: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: REM N=0
1202-PAPER 5: CLS : PRINT AT 10,0; INK 1; OVER
1;"oooooooooooooooooooooooooooooooooooooooo"; GO SUB 9100: GO
SUB 6000
1203-GO SUB 100: GO TO 1203
1300-DATA
0,0,41,255,239,0,0,0,255,127,63,255,127,255,63,127,25
5,255,2,254,252,255,239,0,0,0,255,255,255
1301-RESTORE 1300: FOR f=USR "n" TO USR "p"+7: READ
g: POKE f,g: NEXT f: REM N=0 O=1 P=0
1302-PRINT AT 3,14; INK 2; OVER 1;"1"; INK 3;"i"; INK
2;"o";AT 4,14;"o"; INK 3;"iiii"; INK 2;"o";AT
5,13;"1"; INK 3;"iiiiiiii"; INK 2;"o";AT 6,12;"1";
INK 3;"iiiiiiii"; INK 2;"o";AT 7,12;"1"; INK
3;"iiiiiiii"; INK 2;"o";AT 8,12;"1"; INK
3;"iiiiiiii"; INK 2;"o"
1303-GO SUB 6000: FOR f=9 TO 16: PRINT AT f,12; INK
2;"1"; INK 3;"iiiiiiii"; INK 2;"o": NEXT f: PRINT
AT 10,0; INK 1;"oooooooooooo";AT 10,25;"oooooooo": FOR
f=17 TO 21: PRINT AT f,10; INK 2;"1"; INK
3;"iiiiiiiiiiiiiiiiiiii"; INK 2;"o": NEXT f
1304-IF U(9)=2 THEN LET i$="nincs"
1305-IF U(9)=1 THEN LET i$="megvan"
1306-IF U(9)<>0 THEN PRINT AT 14,25; BRIGHT 1; INK
0;"3": BRIGHT 0: LET q=9
1310-GO SUB 100: GO TO 1310
1400-DATA 0,0,41,255,239,0,0,0
1401-RESTORE 1400: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: REM N=0
1402-PAPER 5: CLS : PRINT AT 10,0; INK 1; OVER
1;"oooooooooooooooooooooooooooooooooooooooo"; GO SUB 9100: GO
SUB 6000
1403-GO SUB 100: GO TO 1403

```

```

1500-DATA 0,0,41,255,239,0,0,0
1501-RESTORE 1500: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: REM N=0
1502-PAPER 5: CLS: PRINT AT 10,0: INK 1; OVER
1;"000000000000000000000000000000000000000000000000000": GO SUB 9100: GO
SUB 6000
1503-GO SUB 100: GO TO 1503
1600-DATA 0,0,41,255,239,0,0,0
1601-RESTORE 1600: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: REM N=0
1602-PAPER 5: CLS: PRINT AT 10,0: INK 1; OVER
1;"000000000000000000000000000000000000000000000000000": GO SUB 9100: GO
SUB 6000
1603-GO SUB 100: GO TO 1603
1700-DATA
0,0,41,255,239,0,0,0,255,127,63,255,127,255,63,127
1701-RESTORE 1700: FOR f=USR "n" TO USR "o"+7: READ
g: POKE f,g: NEXT f: REM N=0 O=1
1702-PAPER 5: CLS: PRINT AT 10,0: INK
1;"000000000000000000000000000000000000000000000000000": FOR f=8 TO 21:
PRINT AT f,28: INK 3;"1"; INK 2;"iii": NEXT f: GO SUB
9100: GO SUB 6000
1703-IF U(4)=2 THEN LET i$="nincs"
1704-IF U(4)=1 THEN LET i$="megvan"
1705-IF U(4)<>0 THEN PRINT AT 12,27: BRIGHT 1: INK
0;"3": BRIGHT 0: LET q=4
1710-GO SUB 100: GO TO 1710
1800-DATA 255,252,254,252,255,252,255,255
1801-RESTORE 1800: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: PAPER 5: CLS: REM N=0
1802-FOR f=0 TO 21: PRINT AT f,0: INK 3;"iii"; INK 2;
OVER 1;"0": NEXT f: GO SUB 6000
1803-LET re=9200
1807-GO SUB re
1810-GO SUB 100: GO TO 1807
1900-PAPER 5: CLS: GO SUB 6000
1901-LET re=8000
1905-GO SUB re
1910-GO SUB 100: GO TO 1902
2000-PAPER 5: CLS: GO SUB 6000
2001-LET re=8000
2005-GO SUB re
2010-GO SUB 100: GO TO 2002
2100-DATA 255,127,63,255,127,255,63,127
2101-RESTORE 2100: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: PAPER 5: CLS: REM N=0
2102-GO SUB 6000: FOR f=0 TO 7: PRINT AT f,12: INK
2;"0"; INK 3;"iiiiiiiiiiiiiiiiiiii": NEXT f: FOR f=8
TO 16: PRINT AT f,27: INK 2;"0"; INK 3;"iii": NEXT
f: FOR f=17 TO 21: PRINT AT f,12: INK 2;"0"; INK
3;"iiiiiiiiiiiiiiiiiiii": NEXT f
2103-LET re=9050
2104-IF U(2)=2 THEN LET i$="nincs"
2105-IF U(2)=1 THEN LET i$="megvan"
2106-IF U(2)<>0 THEN PRINT AT 15,26: BRIGHT 1: INK
0;"3": BRIGHT 0: LET q=2
2110-GO SUB re
2120-GO SUB 100: GO TO 2110
2200-DATA 255,252,254,252,255,252,255,255
2201-RESTORE 2200: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: PAPER 5: CLS
2202-GO SUB 6000: FOR f=0 TO 21: PRINT AT f,0: INK
3;"iii"; INK 2; OVER 1;"1": NEXT f
2203-GO SUB 9000
2204-GO SUB 100: GO TO 2204
2300-PAPER 5: CLS: GO SUB 6000
2305-LET re=9200
2307-GO SUB re
2310-GO SUB 100: GO TO 2307
2400-PAPER 5: CLS: GO SUB 6000
2401-LET re=8050
2405-GO SUB re
2410-GO SUB 100: GO TO 2402
2500-DATA 255,127,63,255,127,255,63,127
2501-RESTORE 2500: FOR f=USR "n" TO USR "n"+7: READ
g: POKE f,g: NEXT f: PAPER 5: CLS

```

```

00050000-   GO SUB 6000: FOR f=0 TO 21: PRINT AT f,28; INK
00050001-   CL 1; INK 3; "iii": NEXT f
00050004-   GO SUB 9000
00050005-   DATA 255,127,63,255,127,255,63,127
00050006-   RESTORE 3000: FOR f=USR "n" TO USR "n"+7: READ
00050007-   f,g: POKE f,g: NEXT f: PAPER 5: CLS
00050008-   GO SUB 6000: FOR f=0 TO 21: PRINT AT f,0; INK
00050009-   "iii"; INK 2; "0"; AT 20,0; INK
00050010-   "iii"; INK 2; "0"; AT 21,0; INK
00050011-   LET re=9000
00050012-   GO SUB re
00050013-   GO SUB 100: GO TO 2705
00050014-   PAPER 5: CLS : GO SUB 6000
00050015-   LET re=8050
00050016-   GO SUB re
00050017-   DATA 255,127,63,255,127,255,63,127
00050018-   RESTORE 2900: FOR f=USR "n" TO USR "n"+7: READ
00050019-   f,g: POKE f,g: NEXT f: PAPER 5: CLS
00050020-   GO SUB 6000: FOR f=0 TO 7: PRINT AT f,12; INK
00050021-   "0"; INK 3; "iiiiiiiiiiiiiiiiiiii": NEXT f: FOR f=8
00050022-   TO 16: PRINT AT f,27; INK 2; "0"; INK 3; "iiii": NEXT
00050023-   f: FOR f=17 TO 21: PRINT AT f,0; INK 2; "0"; INK
00050024-   "iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii": NEXT f
00050025-   LET re=9050
00050026-   IF U(1)=2 THEN LET i#="ninacs"
00050027-   IF U(1)=1 THEN LET i#="megvan"
00050028-   IF U(1)<>0 THEN PRINT AT 15,26; BRIGHT 1; INK
00050029-   "3": BRIGHT 0: LET q=1
00050030-   GO SUB re
00050031-   GO SUB 100: GO TO 2910
00050032-   DATA 255,252,254,252,255,252,255,255
00050033-   RESTORE 3000: FOR f=USR "n" TO USR "n"+7: READ
00050034-   f,g: POKE f,g: NEXT f: PAPER 5: CLS
00050035-   GO SUB 6000: FOR f=0 TO 21: PRINT AT f,0; INK
00050036-   "iii"; INK 2; OVER 1; "1": NEXT f
00050037-   LET re=9200
00050038-   GO SUB re
00050039-   GO SUB 100: GO TO 3005
00050040-   PAPER 5: CLS : GO SUB 6000
00050041-   LET re=8000
00050042-   GO SUB re
00050043-   GO SUB 100: GO TO 3102
00050044-   PAPER 5: CLS : GO SUB 6000
00050045-   LET re=9200
00050046-   GO SUB re
00050047-   GO SUB 100: GO TO 3205
00050048-   DATA 255,127,63,255,127,255,63,127
00050049-   RESTORE 3300: FOR f=USR "n" TO USR "n"+7: READ
00050050-   f,g: POKE f,g: NEXT f: PAPER 5: CLS
00050051-   GO SUB 6000: FOR f=0 TO 21: PRINT AT f,28; INK
00050052-   OVER 1; "0"; INK 3; "iii": NEXT f
00050053-   LET re=8050
00050054-   GO SUB re
00050055-   GO SUB 100: GO TO 3305
00050056-   DATA 255,252,254,252,255,252,255,255
00050057-   RESTORE 3400: FOR f=USR "n" TO USR "n"+7: READ
00050058-   f,g: POKE f,g: NEXT f: PAPER 5: CLS
00050059-   GO SUB 6000: FOR f=0 TO 21: PRINT AT f,0; INK
00050100-   "iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii" AND f<=2; INK
00050101-   "0" AND f<=2 AND f>=0; AT f,0; INK
00050102-   "iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii" AND f>=3 AND f<=5; INK
00050103-   "0" AND f>=3 AND f<=5; AT f,0; INK

```


PIRAMIS SZOLITER-JÁTÉK

A játékban közrefova két másik számot,le kell venni,és ezt folytani kell,míg a játék kockái elfogynak.



```
1-REM Piramis szoliter
2-REM
3-REM Nyitrai 2001'
5-GO TO 9500
8-DIM Z$(15,17): LET 0=0: LET DEMO=0
9-REM dimenzionalas
10-BORDER 0: INK 7: PAPER 0: CLS : DIM I(15)
11-LET L=5: LET 0=1
12-PRINT AT 0,28;"NY.L"
20-DIM L(15,2)
30-FOR X=1 TO 15
40-LET I(X)=1
50-NEXT X
55-REM a szamok kivlasztasa
60-LET Z=INT (1+RND*15)
61-IF DEMO=1 THEN LET Z=5
70-LET I(Z)=0
73-REM DEMO adatok beolvasasa
75-RESTORE
80-FOR X=1 TO 15
100-READ L(X,1),L(X,2)
110-NEXT X
115-REM Koordinatak a DEMO-hoz
120-DATA
13,2,13,6,13,10,13,14,13,18,11,4,11,8,11,12,11,16,9,6
,9,10,9,14,7,8,7,12,5,10
125-GO SUB 8000
140-IF DEMO=1 THEN GO TO 1000
```

```

141-INPUT "Melyik mezorol? ";LINE A#: IF A#="STOP"
OR A#="stop" OR A#="STOP" THEN GO TO 9500
150-INPUT "Hova ugrasz? ";LINE B#: IF B#="STOP" OR
B#="stop" OR B#="STOP" THEN GO TO 9500
151-IF CODE A#<48 OR CODE A#>57 THEN PRINT #1;"A
szak 1-15 lehet !!!";BEEP .50,-10: GO TO 130
152-IF CODE B#<48 OR CODE B#>57 THEN PRINT #1;"A
szak 1-15 lehet !!!";BEEP .50,-10: GO TO 130
154-LET a=VAL a#: LET b=VAL b#
156-IF a>15 OR b>15 THEN PRINT #1;"Csak 1-15 lehet
!!!";BEEP .50,-10: GO TO 130
157-REM bevitt lepes ellenorzes
160-RESTORE 180
170-READ C,D,E
175-IF A=C AND B=D THEN GO TO 205
176-IF C=999 THEN PRINT #1;"Rossz, ismeteld! ";A;"-
";B: BEEP .50,-10: GO TO 140
177-GO TO 170
178-REM JU lepesek
180-DATA
1,3,0,0,0,4,3,3,5,4,6,8,7,7,9,8,10,12,11,3,1,2,4,2,3,5,
3,4,0,0,0,7,9,7,8,12,10,11
190-DATA
1,10,6,6,13,10,10,15,13,2,11,7,7,14,11,3,12,8,10,1,6,
13,6,10,15,10,13,11,2,7,14,7,11,12,3,8
200-DATA
5,12,9,9,14,12,12,15,14,4,11,8,8,13,11,3,10,7,12,5,9,
14,9,10,15,12,14,11,4,8,13,8,11,10,3,7,999,999,999
205-IF I(E)=0 THEN PRINT #1;"Teves lepes! ";A;"-
";B: BEEP .50,-10: GO TO 140
210-IF I(A)=1 AND I(B)=0 AND I(E)=1 THEN LET
I(A)=0: LET I(B)=1: LET I(E)=0: PRINT AT
L(A,1),L(A,2)-1;PAPER 0;"":PRINT AT
L(A,1),L(A,2);PAPER 0;INK 7;A:PRINT AT
L(B,1),L(B,2)-1;PAPER 7;"":PRINT AT
L(B,1),L(B,2);PAPER 7;INK 0;B:PRINT AT
L(E,1),L(E,2)-1;PAPER 0;"":PRINT AT
L(E,1),L(E,2);PAPER 0;INK 7;E:PRINT AT L,23;A;"-
";B: LET L=L+1: BEEP .01,39
211-IF I(E)=1 THEN PRINT #1;"Teves lepes! ";A;"-";B:
BEEP .50,-10
250-REM DEMO-jatek vege
260-RESTORE 180
270-READ C,D,E
280-IF C=999 THEN GO TO 9100
285-IF I(C)=1 AND I(D)=0 AND I(E)=1 AND DEMO=1 THEN
RETURN
290-IF I(C)=1 AND I(D)=0 AND I(E)=1 THEN GO TO 140
300-GO TO 270
1000-REM DEMO
1001-FOR O=1 TO 13
1003-IF O=1 THEN LET A=12: LET B=5
1004-IF O=2 THEN LET A=15: LET B=12
1005-IF O=3 THEN LET A=7: LET B=9
1006-IF O=4 THEN LET A=13: LET B=8
1007-IF O=5 THEN LET A=6: LET B=13
1008-IF O=6 THEN LET A=4: LET B=11
1009-IF O=7 THEN LET A=2: LET B=4
1010-IF O=8 THEN LET A=5: LET B=3
1011-IF O=9 THEN LET A=9: LET B=14
1012-IF O=10 THEN LET A=14: LET B=7
1013-IF O=11 THEN LET A=3: LET B=10
1014-IF O=12 THEN LET A=13: LET B=6
1015-IF O=13 THEN LET A=1: LET B=10
1016-GO SUB 157
1017-NEXT O
7999-REM Jatek mezo
8000-PRINT AT 0,28;"NY.L"
8001-INK 6: PLOT 180,138: DRAW 46,0: DRAW 0,-130:
DRAW -46,0: DRAW 0,130
8002-PLOT 180,138: DRAW 4,4: DRAW 46,0: DRAW -4,-4:
PLOT 230,142: DRAW 0,-130: DRAW -4,-4
8004-PLOT 20,160: DRAW 200,0: DRAW 0,-4: DRAW -200,0:
DRAW 0,4: PLOT 20,160: DRAW 4,4: DRAW 200,0: DRAW -
4,-4: PLOT 224,164: DRAW 0,-4: DRAW -4,-4
8005-PRINT AT 2,3;INVERSE 1;" SZOLITER 2001"

```

```

8007-PLOT 0,0: DRAW 251,0: DRAW 4,4: DRAW -251,0:
DRAW -4,-4
8010-FOR X=1 TO 15
8011-BEEP .01,-1
8020-PRINT AT L(X,1),L(X,2)-1; PAPER 7;"   ": PRINT
AT L(X,1),L(X,2); PAPER 7; INK 0;X
8030-NEXT X
8035-PRINT AT L(Z,1),L(Z,2)-1; PAPER 0;"   ": PRINT
AT L(Z,1),L(Z,2); PAPER 0; INK 7;Z
9000-FOR a=0 TO 72 STEP 16
9000-FOR p=7+a TO 135-a STEP 32
9001-BEEP .01,1
9010-PLOT p,72+a: DRAW 25,0: DRAW 0,-9: DRAW -25,0:
DRAW 0,9
9011-PLOT p,72+a: DRAW 4,4: DRAW 25,0: DRAW -4,-4:
PLOT p+29,76+a: DRAW 0,-9: DRAW -4,-4
9020-NEXT p
9030-NEXT a
9035-PRINT #1; FLASH 1;" Bemutato-jatek."
9040-RETURN
0100-REM lepes
0110-LET M=0
0115-FOR X=1 TO 15
0120-IF I(X)=1 THEN LET M=M+1
0125-NEXT X
0130-PRINT AT 17,0;"maradt meg:";M;" lepes"
0132-PRINT "ertekes:";
0135-IF M=1 THEN PRINT INVERSE 1;"szolimester"
0140-IF M=2 THEN PRINT BRIGHT 1;"mester!!"
0145-IF M=3 THEN PRINT "felprofi!"
0150-IF M=4 THEN PRINT FLASH 1;"amator!!! "
0160-IF M>=5 THEN PRINT FLASH 1;"bena-kacsa"
0185-PLOT 0,14: DRAW 169,0: DRAW 0,26: DRAW -169,0:
DRAW 0,-26: PLOT 0,14+26: DRAW 4,4: DRAW 169,0: DRAW
-4,-4: PLOT 173,18+26: DRAW 0,-26: DRAW -4,-4
0186-RESTORE 9188
0187-IF M<=2 THEN FOR X=1 TO 15: READ A,B: BEEP
B/100,A+12: NEXT X
0188-DATA
V,10,7,10,7,15,2,15,11,10,11,10,11,15,7,15,7,10,11,10
0,14,26,14,10,12,10,11,10,9,25
0200-IF NOT M=1 THEN PRINT #1;" Ezt meg lehet
gyakorolni!": PAUSE 1: PAUSE 0: GO SUB 9900: GO TO
0235
0210-IF DEMO=1 THEN GO SUB 9900: GO TO 9235
0211-INPUT "IRD be a neved:";LINE Z$(q)
0225-REM Menü
0230-GO SUB 9900
0232-LET Q=Q+1: IF Q=12 THEN LET Q=8
0233-LET DEMO=0
0235-LET DEMO=0: BEEP .01,30: BEEP .020,31: BEEP
0.030,32: IF CODE INKEY#=49 THEN GO TO 9600
0240-IF CODE INKEY#=50 THEN GO TO 9
0245-IF CODE INKEY#=51 THEN LET DEMO=1: GO TO 9
0246-IF CODE INKEY#=48 THEN GO TO 9999
0250-GO TO 9235
0500-DIM Z$(15,17): POKE 23609,1
0501-RESTORE 9560
0510-FOR M=1 TO 3
0520-READ X#
0530-FOR N=0 TO 7
0540-READ X
0550-POKE USR X#+N,X
0551-NEXT N
0552-NEXT M
0560-DATA "4",8,16,68,198,198,198,124,0
0570-DATA "K",16,32,112,48,48,48,120,0
0580-DATA " ",8,16,124,198,198,198,124,0
0590-LET Q=0: GO SUB 9900: GO TO 9235
0591-GO SUB 9900: GO TO 9235
0599-REM Jatekismerteto
0600-BORDER 0: PAPER 0: CLS : PRINT AT 0,28;"NY.L"
0602-PLOT 20,160: DRAW 200,0: DRAW 0,-4: DRAW -200,0:
DRAW 0,4: PLOT 20,160: DRAW 4,4: DRAW 200,0: DRAW -
4,-4: PLOT 224,164: DRAW 0,-4: DRAW -4,-4
0604-PRINT AT 2,3; INVERSE 1;" S Z O L I T E R
jatek"

```

```

00605-PRINT AT 4,6;"Jatek-ismerteto:"
00610-PRINT AT 5,6;"=====
00620-PRINT AT 7,4;" A jatek lenyege:";AT 8,4;" mindig
az urres kocara"
00630-PRINT AT 9,4;" kell ugorni,atugorva a";AT 10,4;"
koztelevo masikat."
00640-PRINT AT 11,4;" Az ugras lehet ferden,";AT
12,4;" vizszintesen, le,es fel."
00650-PRINT AT 13,4;" Ezt addig kell folytatni,";AT
14,4;" mig lehetoseg van erre."
00660-PRINT AT 16,4;" A STOP-beirasaval a jatekot";AT
17,4;" meg lehet allitani."
00700-PRINT AT 19,1;" Nyomj egy gombot!": PAUSE
1: PAUSE 0
00800-GO TO 9591
00899-REM PROGRAM-inditsa
00900-PAPER 1: BORDER 0: CLS
00910-PRINT AT 0,28;"NY.L"
00950-INK 6: PLOT 20,160: DRAW 200,0: DRAW 0,-4: DRAW
-200,0: DRAW 0,4: PLOT 20,160: DRAW 4,4: DRAW 200,0:
DRAW -4,-4: PLOT 224,164: DRAW 0,-4: DRAW -4,-4
00960-PRINT AT 2,3; PAPER 1; INK 6; INVERSE 1;" S Z O
L H I T E R 2001"
00970-PRINT AT 4,1; PAPER 1; INK 6;" © Nyitrai
Laszlo";AT 5,10;"MMI ev majusaban"
00983-PRINT #0;" Nyomd a megfelelo szamot!"
00985-PRINT AT 8,3; PAPER 7; INK 2; INVERSE 1;"Jatek
szoliter-mestereknek"
00986-PRINT AT 12,6; PAPER 1; INK 6;"1=jatek-
magyarazata."
00987-PRINT AT 14,6; PAPER 1; INK 6;"2=jatek-
inditasa."
00988-PRINT AT 16,6; PAPER 1; INK 6;"3=jatek-
beutatoja."
00989-PRINT AT 18,6; PAPER 1; INK 6;"0=kilepes a
jatekbol!"
00990-PLOT 0,0: DRAW 255,0: DRAW 0,168: PLOT 223,175:
DRAW -223,0: DRAW 0,-175
00995-RETURN
00996-REM SAVE RUTIN
00999-INK 0: PAPER 7: BORDER 7: CLEAR : PRINT AT
10,1;"A jatek 'RUN'-nal ujraindithato"

```

LÁDATOLOGATÓ JÁTÉK (SOKOBAN)

Ládatologató játék Graphics BASIC-ban készítve:Itt egy raktárost kell irányítani,akinek az a feladata,hogy a nagy ládákat a kijelölt helyükre kell tolni,kikerülve az akadályokat,és ezeket a megadott időn belül el kell végezni.A pálya teljesítése után automatikusan a következő pályára ugrik.Jelenleg 15 pálya van,de a játékszerkesztővel még újabb pályákat is lehet készíteni.



```
1-REM NYITRAI 2000'
2-REM
3-REM LADATOLOGATO -SOKOBAN- JATEK
4-REM
5-REM HIBA ESETEN                                RANDOMIZE
USR 54500
6-REM
7-GO TO 5050
9-LET PH=15
10-GO SUB 5100: DIM K(7): RESTORE 20: FOR I=1 TO 7:
READ K(I): NEXT I
20-DATA 19,20,21,11,38,33,40
30-LET K$="1234567890QWERTYUIOPASDFGHJKL"+CHR$
0+CHR$ 0+"ZXCVBNM"+CHR$ 0+" "
40-DIM R$(10,10): DIM R(10)
50-FOR I=1 TO 10: LET R$(I)="NYITRAI": LET
R(I)=(11-I)*100: NEXT I
55-RESTORE 90: DIM S(3,8): FOR I=1 TO 3: FOR J=1 TO
8: READ S(I,J): NEXT J: NEXT I
90-DATA
10,11,10,9,12,13,12,9,15,16,15,14,17,18,17,14,20,21,2
0,19,22,23,22,19
100-_BRON :_OFFERR :INK 6: POKE 23658,8: BORDER 1:
PAPER 1: CLS
110-_COLOR 1,8;4,18;104:_COLOR 0,7;4,18;120
120-_ATOFF :_SIZE 1,2;12:_SPRINT
56,1;4,1;"Ládatologató":_SPRINT 85,15;4,1;" Játék"
```



```

1040- _DIRECT 1;-2,0:_DIRECT 2;2,0:_DIRECT
3;0,2:_DIRECT 4;0,-2
1050-GO SUB 5000: LET ID=DB*70: GO SUB 5300
1060- _PLACE 1;16*XX,16*YY:LET R=1
1090- _FAST :LET HP=15: LET S=5:_ONERR 1400
1095-POKE 23672,0
1100-GO SUB 5200: IF B+J+F+L<>1 THEN GO TO 1400
1110-GO TO 1050+(B+2*J+3*L+4*F)*70
1120-IF R<>1 THEN _PLACE 1;XX*16,YY*16:_SET R:_SET 1:
LET R=1
1125-LET A=P(YY+1,XX): IF A>3 THEN GO TO 1150
1130-FOR I=1 TO 8:_SOUND S;0,0;100,100: GO SUB
5600:_MOVE 1: NEXT I: LET XX=XX-1
1135-LET B=?KEY(K(1)): IF B=1 AND ID>0 THEN GO TO
1125
1140-GO TO 1400
1150-IF A>5 THEN GO TO 1400
1155-LET B=P(YY+1,XX-1): IF B>3 THEN GO TO 1400
1160- _INTO U(YY+1,XX-1)TO 25;0,0;0:_INTO U(YY+1,XX)
TO 25;16,0;0:_INTO U(YY+1,XX+1) TO 25;32,0;0
1165-FOR I=1 TO 8: GO SUB 5600:_INTO 25 TO
26;0,0;0:_INTO A TO 26;16-I*2,0;0:_ATOFF :_INTO
S(1,I) TO 26;32-I*2,0;3:_ATON :_PUT 26;(XX-
2)*16,YY*16;0: NEXT I
1170-LET P(YY+1,XX-1)=A: LET P(YY+1,XX)=1: LET XX=XX-
1
1173- _PLACE 1;XX*16,YY*16
1175-LET B=?KEY(K(1)): IF B=1 AND ID>0 THEN GO TO
1155
1180-GO TO 1400
1190-IF R<>2 THEN _PLACE 2;XX*16,YY*16:_SET R:_SET 2:
LET R=2
1195-LET A=P(YY+1,XX+2): IF A>3 THEN GO TO 1220
1200-FOR I=1 TO 8:_SOUND S;0,0;100,100: GO SUB
5600:_MOVE 2: NEXT I: LET XX=XX+1
1205-LET J=?KEY(K(2)): IF J=1 AND ID>0 THEN GO TO
1195
1210-GO TO 1400
1220-IF A>5 THEN GO TO 1400
1225-LET B=P(YY+1,XX+3): IF B>3 THEN GO TO 1400
1230- _INTO U(YY+1,XX+1)TO 25;0,0;0:_INTO U(YY+1,XX+2)
TO 25;16,0;0:_INTO U(YY+1,XX+3) TO 25;32,0;0
1235-FOR I=1 TO 8: GO SUB 5600:_INTO 25 TO
26;0,0;0:_INTO A TO 26;16+I*2,0;0:_ATOFF :_INTO
S(2,I) TO 26;I*2,0;3:_ATON :_PUT 26;XX*16,YY*16;0:
NEXT I
1240-LET P(YY+1,XX+3)=A: LET P(YY+1,XX+2)=1: LET
XX=XX+1
1243- _PLACE 2;XX*16,YY*16
1245-LET J=?KEY(K(2)): IF J=1 AND ID>0 THEN GO TO
1225
1250-GO TO 1400
1260-IF R<>3 THEN _PLACE 3;XX*16,YY*16:_SET R:_SET 3:
LET R=3
1265-LET A=P(YY+2,XX+1): IF A>3 THEN GO TO 1290
1270-FOR I=1 TO 8:_SOUND S;0,0;0,0:_MOVE 3: GO SUB
5600: NEXT I: LET YY=YY+1
1275-LET L=?KEY(K(3)): IF L=1 AND ID>0 THEN GO TO
1265
1280-GO TO 1400
1290-IF A>5 THEN GO TO 1400
1295-LET B=P(YY+3,XX+1): IF B>3 THEN GO TO 1400
1300- _INTO U(YY+1,XX+1)TO 27;0,0;0:_INTO U(YY+2,XX+1)
TO 27;0,16;0:_INTO U(YY+3,XX+1) TO 27;0,32;0
1305-FOR I=1 TO 8: GO SUB 5600:_INTO 27 TO
28;0,0;0:_INTO A TO 28;0,16+I*2;0:_ATOFF :_INTO
S(3,I) TO 28;0,I*2;3:_ATON :_PUT 28;XX*16,YY*16;0:
NEXT I
1310-LET P(YY+3,XX+1)=A: LET P(YY+2,XX+1)=1: LET
YY=YY+1
1313- _PLACE 3;XX*16,YY*16
1315-LET L=?KEY(K(3)): IF L=1 AND ID>0 THEN GO TO
1295
1320-GO TO 1400
1330-IF R<>4 THEN _PLACE 4;XX*16,YY*16:_SET R:_SET 4:
LET R=4
1335-LET A=P(YY,XX+1): IF A>3 THEN GO TO 1360

```



```

1340-FOR I=1 TO 8:_SOUND S;0,0;100,100: GO SUB
5600:_MOVE 4: NEXT I: LET YY=YY-1
1345-LET F=?KEY(K(4)): IF F=1 AND ID>0 THEN GO TO
1355
1350-GO TO 1400
1360-IF A>5 THEN GO TO 1400
1365-LET B=P(YY-1,XX+1): IF B>3 THEN GO TO 1400
1370-_INTO U(YY-1,XX+1) TO 27;0,0;0:_INTO U(YY,XX+1)
TO 27;0,16;0:_INTO U(YY+1,XX+1) TO 27;0,32;0
1375-FOR I=1 TO 8: GO SUB 5600:_INTO 27 TO
28;0,0;0:_INTO A TO 28;0,16-I*2;0:_ATOFF :_INTO
S(3,I) TO 28;0,32-I*2;3:_ATON :_PUT 28;XX*16,(YY-
2)*16;0: NEXT I
1380-LET P(YY-1,XX+1)=A: LET P(YY,XX+1)=1: LET YY=YY-
1
1393-_PLACE 4;XX*16,YY*16
1395-LET F=?KEY(K(4)): IF F=1 AND ID>0 THEN GO TO
1385
1400-IF B+J+L+F=0 THEN FOR I=1 TO DB: IF
P(D(I,2),D(I,1))=5 THEN IF INKEY$="" THEN NEXT I: GO
TO 1500
1410-GO SUB 5600
1420-LET B=?KEY(K(7)): LET A=?KEY(K(6)): GO SUB 5325:
IF ID=0 THEN GO TO 1700
1425-IF A=1 THEN GO TO 1730
1426-IF B=1 THEN GO TO 100
1430-GO TO 1100
1500-REM UJ PALYA
1505-LET PN=PN+ID*10
1510-_SCROLL SCREEN# 0,3,24
1520-LET PA=PA+1: IF PA<PH+1 OR GS>2000 THEN GO TO GS
1530-PAPER 6: BORDER 6: INK 1: CLS
1540-_COLOR 4,4;15,25;88:_COLOR 3,3;15,25;120
1550-_ATOFF :_SIZE 1,1;7
1560-_SPRINT 32,32;0,1;"GRATUL@LOK !"
1570-_SPRINT 32,56;0,1;"Teljes@tettet a ?sszes "
1575-_SPRINT 32,80;0,1;"nehozsogi szintet. Rajtad"
1580-_SPRINT 32,80;0,1;"nem tudtam kifogni."
1585-_SPRINT 32,100;0,1;"Ysszesen "+STR$ pn+"
Pontot@"
1590-_SPRINT 32,112;0,1;"szereztol."
1600-GO TO 1820
1700-BORDER 5: PAPER 5: INK 0: CLS
1710-PAUSE 1:_SIZE 1,2;8
1720-_COLOR 7,6;4,22;88:_COLOR 6,5;4,22;120
1725-_ATOFF :_SPRINT 52,56;0,1;"Elfogyott az id@d !"
1730-LET EL=EL-1: IF EL=0 THEN GO TO 1800
1740-PAUSE 10: PAUSE 300: GO TO GS
1800-_COLOR 11,6;4,22;88:_COLOR 10,5;4,22;120
1810-_SPRINT 52,88;0,1;"A j@toknak voge !"
1820-FOR i=10 TO 1 STEP -1: IF pn>R(i) THEN NEXT I
1825-LET I=I+1
1830-IF I=11 THEN GO TO 100
1840-_ONERR 1860
1850-FOR J=10 TO I: LET R$(J)=R$(J-1): LET R(J)=R(J-
1): NEXT J
1860-INPUT "urd be a neved : ";r$(i)
1865-_OFFERR :LET R(I)=PN
1870-GO TO 200
2000-REM TERVEZO
2010-_ATON :GO SUB 5100
2020-_SIZE 1,1;10:_SPRINT 38,8;64,0;"L@DATOLOGAT@-
U@TUK@"
2030-_SIZE 1,1;8:_SPRINT 76,64;4,0;"P@lya
te"+("rvez@" AND TV=2050)+("sztel@" AND TV<>2050)
2040-_ONERR 2040:INPUT "Sz@ma (1-30) :";PA
2043-IF PA<0 OR PA>30 OR PA<>INT PA THEN GO TO 2040
2044-IF PA=0 THEN GO TO 100
2045-_OFFERR :GO TO TV
2050-LET M=0: GO SUB 5000
2060-LET V=1: FOR I=0 TO 7:_PUT I+1;I*16,176;0: NEXT
I
2065-_COLOR 22,0;2,2;199:_SIZE 1,1;5:_SPRINT
132,180;0,1;"T?r?l M@sik Kosz Teszt"
2070-FOR i=0 TO 3:_COLOR 22,16+I*4;2,4;96+I*8: NEXT I
2075-LET ST=0:_SCALE 16;24:_PLACE 16;120,80:_MODE
16;1,0:_SET 16

```

```

2100-GO SUB 5200: IF B+J+L+F+T<>0 THEN GO TO 2110
2101-LEFT ST=0
2102-LET A=?KEY(K(6)): LET B=?KEY(K(7)): IF A+B=0
THEN GO TO 2100
2103-GO TO 100
2110-LEFT ST=ST+(ST<8): _DIRECT 16;ST*(J-B),ST*(L-F)
2120-MOVE 16:IF T=0 THEN GO TO 2100
2125-LET Y=?YPOS(16): LET Y=Y+7
2130-IF Y>175 THEN GO TO 2150
2135-LET X=?XPOS(16): LET Y=INT(Y/16): LET X=INT
((X+7)/16): _SET 16:_PUT U;X*16,Y*16;0:_SET 16: LET
P(Y+1,X+1)=U: GO TO 2100
2150-LET X=?XPOS(16): LET X=X+7
2155-IF X>127 THEN GO TO 2190
2160-_SET 16:_PUT U;(U-1)*16,176;0:LET U=INT
(X/16)+1: LET A=ATTR(22,(U-1)*2): _COLOR 22,(U-
1)*2;2,2;A+128:_SET 16: GO TO 2100
2190-PUT U;(U-1)*16,176;0:_SET 16:GO TO 2200+INT
((X-128)/32)*50
2200-FOR Y=1 TO 11: FOR X=1 TO 16: LET P(Y,X)=1: NEXT
X: NEXT Y
2210-GO SUB 5100: GO TO 2060
2250-GO SUB 2400: IF A=0 THEN GO TO 2000
2260-SIZE 1,2;7:_SPRINT 0,176;0,0;"Hib@s p@ly@t
tervez! Javitsd ki!"
2270-PAUSE 1: PAUSE 0
2275-SIZE 1,2;7:_SPRINT 0,176;0,3;"Hib@s p@ly@t
tervez! Javitsd ki!"
2280-GO TO 2060
2300-GO SUB 2400: IF a=1 THEN GO TO 2255
2310-GO TO 100
2350-GO SUB 2400: IF A=1 THEN GO TO 2255
2360-LET M=3: LET GS=2370: LET EL=2: LET PN=0
2365-GO TO 1010
2370-GO TO 100
2400-LET CIM=45000+(PA-1)*176: DIM T(8): FOR Y=1 TO
11: FOR X=1 TO 16: LET A=P(Y,X): POKE CIM,A: LET
CIM=CIM+1: LET T(A)=T(A)+1: NEXT X: NEXT Y
2410-LET A=0: IF T(2)<>T(5) OR T(2)>14 OR T(5)>14
THEN LET A=1
2420-IF T(8)<>1 THEN LET A=1
2430-RETURN
5000-REM PALYA RAJZOLAS
5005-_ATON :GO SUB 5100:_COLOR 0,0;22,32;8
5010-LET DB=0: DIM D(14,2): DIM P(11,16): DIM
U(11,16): LET CIM=45000+(PA-1)*176
5020-FOR Y=0 TO 10: FOR X=0 TO 15: LET A=PEEK
CIM:_PUT A;X*16,Y*16;0: LET CIM=CIM+1
5030-LET U(Y+1,X+1)=A-((A-1) AND (A=4 OR A=5 OR
A=8)): LET P(Y+1,X+1)=A-((A-1) AND (A=2 OR A=3 OR
A=8) AND M<>0): IF A=2 THEN LET DB=DB+1: LET
D(DB,1)=X+1: LET D(DB,2)=Y+1
5035-IF A=8 THEN LET XX=X: LET YY=Y
5040-NEXT X: NEXT Y
5045-RETURN
5050-REM MAGYARAZAT
5051-BORDER 1: PAPER 1: INK 2: CLS
5052-LET A#=CHR$ 128+CHR$ 132+CHR$ 136+CHR$ 0+CHR$ 4
5054-RESTORE 5110
5056-READ a: IF a=0 THEN GO TO 5066
5057-FOR I=3 TO 12:_PUT 8;I*16,80;0: NEXT I
5058-READ B,D,E,B#
5060-SIZE A,B;0*A
5062-FOR I=1 TO 5: PAUSE 5:_SPRINT D,E;CODE
I$(I),0;B#: INK 2: NEXT I
5063-FOR I=3 TO 12:_PUT 5;I*16,80;0: NEXT I
5064-GO TO 5056
5066-_ATON :FOR I=-1.5 TO 1.5:_PUT 3+ABS
H;120+I*32,144;0: NEXT I
5068-FOR I=3 TO 12:_PUT 6;I*16,160;0: NEXT I
5070-PAUSE 150:_BRON :_OFFERR : INK 5:_SCROLL SCREEN#
0,4,32
5072-RESTORE 5074: PAUSE 100: LET A=17: GO SUB 5124
5074-DATA " Ebben a j@tokban egy rakt@rost kell he-"
5076-DATA " lyettesitened,aki az egyes szinteken lo-"
5078-DATA "v@ l@d@kat a kijel?lt helyekre tolja. "
5079-DATA "A j@tokban a nehozsoget az adja, hogy "

```

```

500000- DATA "csak elkre lehet mozogni, os noha egy-"
500001- DATA "egy kkolonc is akadalyozza utunkat,"
500002- DATA "bez ezeket el lehet az 3tbl tolni."
500003- DATA "Ajtok palyaszertkvel is el van"
500004- DATA "sltva, igy sajat szintet is lehet ko-"
500005- DATA "A nitei program szalagra, lemezre ment."
500006- DATA "A program j ldatologatst kvan."
500007- DATA " "Sznestechnika: LM1000 chip", "
500008- DATA " "L a chip", " Zene : 2554-es port."
500009- DATA " NYITRAI L5ZLh 2000."
500010- REM CLS
500011- BOARDER 0: PAPER 0: INK 7: BRIGHT 1: FLASH 0:
500012- INVERSE 0: OVER 0: CLS: RETURN
500013- _SIZE 1,1,120,16,"AZ"
500014- _SIZE 2,2,64,40,"NYL"
500015- _SIZE 1,1,120,40,"SZOFTVER"
500016- _SIZE 1,1,96,64,"BEMUTATJA"
500017- _SIZE 1,2,64,104,"a NYITRAI SOKO"
500018- _SIZE 1,1,60,132," JUTUK"
500019- DIM A$(40): WINDOW 1;0,8;32,184
500020- _SOUND 1;0;255,255: DRUM 200,1;200,255
500021- FOR I=1 TO 8: FOR J=1 TO 8: READ A$(I,J)
500022- FOR K=0 TO 9: INK K: _SPRINT K*24+8,184-
K;0,1;A$(K*4+1 TO K*4+4): _SCROLL POINT 1,3,1: INK 9-
K: NEXT K
500023- NEXT I: PAUSE 200: GO TO 8
500024- REM BILLENTYUZET
500025- LET B=?KEY(K(1)): LET J=?KEY(K(2)): LET
L=?KEY(K(3)): LET F=?KEY(K(4)): LET T=?KEY(K(5)):
RETURN
500026- REM ADAT TABLA
500027- _SIZE 1,1;8:PAPER 3:_SPRINT 0,176;4,0;" Szint
pont uletek Idk
500028- PAPER 0:_SPRINT 36-(LEN STR$ PA)*4,184;4,0;STR$
PA
500029- _SPRINT 96-(LEN STR$ PN)*4,184;4,0;STR$ PN
500030- _SPRINT 160-(LEN STR$ EL)*4,184;4,0;STR$ EL
500031- _SPRINT 212-(LEN STR$ ID)*4,184;4,0;" "+STR$
ID+"
500032- COLOR 23,0;1,32;71
500033- RETURN
500034- REM ORA LEPTETES
500035- LET V=PEEK 23672: IF V<50 THEN RETURN
500036- LET UV=INT (V/50): POKE 23672,V-UV*50
500037- LET ID=ID-UV: IF ID<0 THEN LET ID=0
500038- RETURN
500039- REM DISC SAVE
500040- CLEAR : LET A=?ADDR(0): _DPOKE 65534,A: SAVE
*"GAME"LINE 9920: SAVE *"chars2"CODE 65368,168: SAVE
*"gamecd"CODE A,54500-A: SAVE *"levels/n"CODE
45000,5280
500041- $STOP
500042- REM DISC LOAD
500043- CLEAR 44999: LOAD *"gamecd"CODE : LOAD
*"basic/sp"CODE : LOAD *"chars2"CODE : LOAD
*"levels/n"CODE : RANDOMIZE USR 54500
500044- LET A=?DPEEK(65534): _START A: RUN
500045- REM LOAD
500046- CLEAR 44999: LOAD ""CODE : LOAD ""CODE :
RANDOMIZE USR 54500: RUN
500047- REM SAVE
500048- CLEAR : SAVE "GAME"LINE 9990: LET A=?ADDR(0):
SAVE "GAME CODE"CODE A,65536-A: SAVE "LEVELS"CODE
45000,5280: CLEAR : PRINT " VERIFY ": VERIFY "":
VERIFY ""CODE : VERIFY ""CODE

```

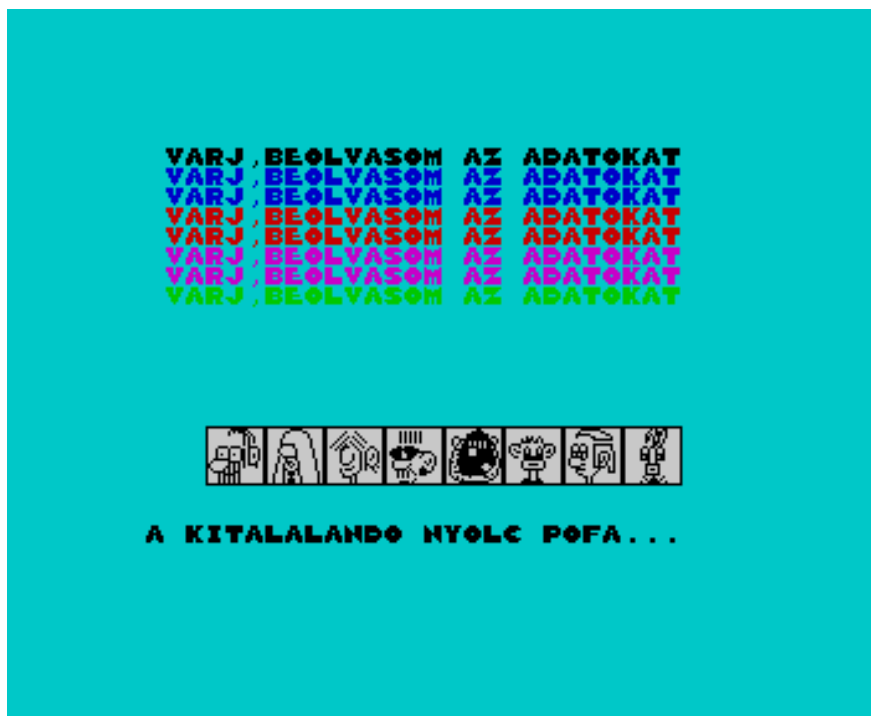
MASTERMIND 128K

Itt van a MASTERMIND 128K új basic játékprogramom,mely más a megszokott mastermind játékoktól,itt 8 POFA néz velünk szemben,és a Spectrum fordulóként 4-et rejt el,melyet találgatásokkal kell megtalálnunk.Minden találgatás után visszajelzést kapunk,melyszerint ha,

1. fehér,akkor egyet sem találtunk el,
2. piros,akkor a négy POFA valamelyike szerepel,de nem ott áll,ahol kiválasztottuk,és végül ha ,
3. sárga,akkor eltaláltuk és a sorban jó helyen is van.

Nincs az korlátozva,hogy mindnek egyszerre ugyanazt válasszuk (mindnek pl. 1-es jelölünk,de akkor a Spectrum téves jelzést adhat)

A POFA-k nem sprite-ban vannak megrajzolva,hanem más karakterkészletben és a megfelelő POKE -kal van beállítva.(9x8=72 karakter) A betűk sem a megszokott – egyszerű – Spectrum karakterkészletek,hanem az ATIC-ATAC játék betűi.



```
0 REM JATEKGYAROS-NYITRAISOFT
5 REM MASTERMIND JATEK 2004
10 RESTORE 15: FOR f=0 TO 7: READ a: POKE USR
"A"+f,a: NEXT f: CLEAR 64600: GOSUB 9000
15 DATA 0,24,60,126,126,60,24,0
20 BORDER 2: PAPER 2: INK 1: CLS : PRINT AT
15,10;"MASTERMIND": PRINT AT 16,10;"128K BASIC"
90 PRINT AT 17,18: PAPER 2: INK 1;"JATEK";AT
19,15;"NYITRAISOFT 2004"
100 FOR n=1 TO 4: t"T240UX1000W007N2c#f#gEA": NEXT
n
105 LET rec=12
110 PRINT #1;" NYOMD LE AZ ENTER-T"
```

```

130 PAUSE 0
1000 REM MENU
1001 POKE 23606,0: POKE 23607,246
1005 GOSUB 8100: BORDER 5: PAPER 5: INK 0: CLS :
PRINT AT 7,9;"MAGYARAZAT (M)";AT 10,9;"JATEK (J)"
1010 LET c#=INKEY$: IF c# <> "m" AND c# <> "j" THEN
GOTO 1010
1020 IF c#="m" THEN GOTO 1500
1030 IF c#="j" THEN GOTO 2000
1500 REM MAGYARAZAT
1510 BORDER 1: PAPER 1: INK 6: CLS
1515 POKE 23606,0: POKE 23607,246
1520 PRINT AT 2,10;"MASTERMIND"
1525 FOR n=1 TO 2:t"t240UX1000U007N2c#f#gEA": NEXT
n
1530 PRINT AT 6,2;"EBBEN A JATEKBAN A RENDELKEZESRE
ALLO NYOLC POFABOL A SPECTRUM NEGYET REJT EL,ES
EZEKET KELL TALALGATASSAL KITALALNI."
1540 PRINT AT 12,2;"MINDEN PROBALKOZASUNK UTAN VA-
LASZT KAPUNK A SZAMITOGEPTEL,
KOVETKEZOK:"
1545 PRINT #1;" NYOMD LE AZ ENTER-T": GOSUB
8100
1547 PAUSE 0: CLS
1550 PRINT AT 3,2;"HA FEHER A JELZES,AZ AZT
JELENTI,HOGY EGY POFA SINCS A KIUA-LASZTOTT
NEGYBOL."
1560 PRINT AT 6,2;"HA PIROS,AZ AZT JELENTI,HOGY
ELTALALTUK A POFAT,DE ROSSZ HELYEN VAN,HA SARGA AKKOR
ELTALALTUK ES JO HELYEN IS ALL."
1565 PRINT AT 11,2;"A JATEKBAN OSSZESEN 12 PROBAL
KOZASUNK VAN,JATSHZATUNK MASIK JATEKOSSAL IS,NEMCSAK
A SPECTRUM ELLEN."
1570 PRINT AT 18,11; PAPER 6; INK 2;"SOK SIKERT!!":
GOSUB 8100
1572 PRINT #1;" NYOMD LE AZ ENTER-T"
1580 PAUSE 0: GOTO 1000
2000 REM kiválasztás
2010 BORDER 0: PAPER 0: INK 7: CLS : PRINT AT
8,3;"JATEKOS A SPECTRUM ELLEN (1)": GOSUB 8100
2020 PRINT AT 11,3;"JATEKOS JATEKOS ELLEN (2)"
2030 LET a#=INKEY$: IF a# <> "1" AND a# <> "2" THEN
GOTO 2030
2040 IF a#="1" THEN LET jug=1
2050 IF a#="2" THEN LET jug=2
2100 REM BEALLITAS
2150 LET int=0: LET aci1=0: LET aci2=0: LET aci0=4
2200 LET v=22: LET z=2: LET e=23: LET r=3: LET s=7:
LET t=11: LET u=15
2210 BORDER 4: PAPER 4: INK 0: CLS : GOSUB 8100
2220 PRINT AT 0,0; PAPER 6; INK
0;"#PROBALKOZAS:";int;" UEGE ";rec;" #"
2222 LET c#=STR$ rec: IF LEN c#<2 THEN PRINT AT
0,0; PAPER 6; INK 0;" ";rec;" #"
2225 PRINT AT 0,26; PAPER 6; INK 0;"MASTER";AT
1,26;" MIND "
2228 PRINT AT 1,0; PAPER 6; INK 0;" "
2230 PRINT AT 1,2; PAPER 2; INK 7;"t6 5 4 3
2
2240 FOR n=0 TO 23: PRINT AT 2,2+n; PAPER 0; INK
7;"t": NEXT n
2245 FOR n=3 TO 17: PRINT AT n,1; INK 1;"t": NEXT n
2250 FOR n=3 TO 17: PRINT AT n,26; INK 1;"t": NEXT
n
2255 LET nn=3: GOSUB 2260: GOTO 2265
2260 PRINT AT nn,26;"ttt";AT nn+1,26;"ttt";AT
nn+2,26;"ttt": RETURN
2265 LET nn=7: GOSUB 2260
2270 LET nn=11: GOSUB 2260
2275 LET nn=15: GOSUB 2260
2280 FOR n=3 TO 23 STEP 4: PRINT AT r,n;"ttt";AT
s,n;"ttt";AT t,n;"ttt";AT u,n;"ttt": NEXT n
2290 IF r=5 THEN GOTO 2300
2295 LET r=r+1: LET s=s+1: LET t=t+1: LET u=u+1:
GOTO 2280
2300 LET r=3: LET s=7: LET t=11: LET u=15

```



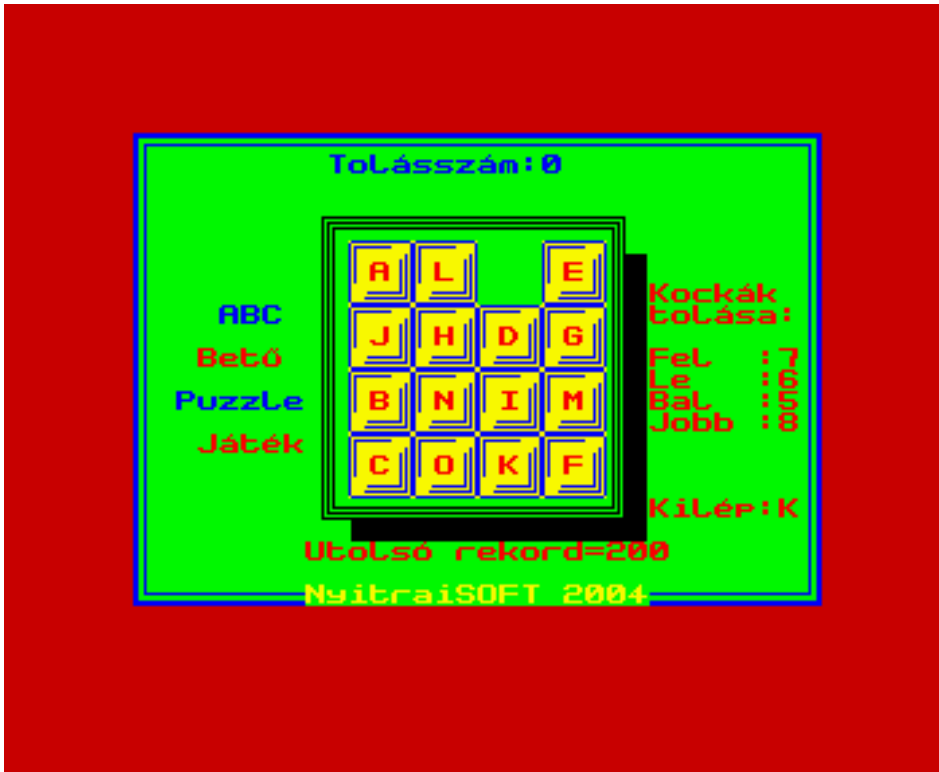
```

3120 GOSUB 8100: INPUT "NEGYEDIK POFA ";m: IF m<1
OR m>8 THEN GOTO 3120
3125 LET li=u: LET co=e: LET kj=m: GOSUB 3015
32000 REM ELLENORZES
32005 POKE 23607,246: PRINT #1; PAPER 4; INK 0;"A
SPECTRUM ERTEKELI A VALASZT": PAUSE 50
3207 FOR n=1 TO 2:t"t240UX1000W007N2c#f#gEA": NEXT
n
3210 IF j=a THEN GOSUB 4000: GOTO 3230
3215 IF j=b THEN GOSUB 4100: GOTO 3230
3220 IF j=c THEN GOSUB 4100: GOTO 3230
3225 IF j=d THEN GOSUB 4100: GOTO 3230
3230 IF k=a THEN GOSUB 4100: GOTO 3250
3235 IF k=b THEN GOSUB 4000: GOTO 3250
3240 IF k=c THEN GOSUB 4100: GOTO 3250
3245 IF k=d THEN GOSUB 4100: GOTO 3250
3250 IF l=a THEN GOSUB 4100: GOTO 3270
3255 IF l=b THEN GOSUB 4100: GOTO 3270
3260 IF l=c THEN GOSUB 4000: GOTO 3270
3265 IF l=d THEN GOSUB 4100: GOTO 3270
3270 IF m=a THEN GOSUB 4100: GOTO 3290
3275 IF m=b THEN GOSUB 4100: GOTO 3290
3280 IF m=c THEN GOSUB 4100: GOTO 3290
3285 IF m=d THEN GOSUB 4000
3290 LET vj=v
3295 IF aci1=0 THEN GOTO 3300
3307 FOR n=1 TO aci1: PRINT AT z,vj; PAPER 0; INK
6;t: LET vj=vj+1: GOSUB 8100: NEXT n
3310 IF aci2=0 THEN GOTO 3320
3315 FOR n=1 TO aci2: PRINT AT z,vj; PAPER 0; INK
2;t: LET vj=vj+1: GOSUB 8100: NEXT n
3320 LET aci0=4-(aci1+aci2): IF aci0=0 THEN GOTO
3500
3330 FOR n=1 TO aci0: PRINT AT z,vj; PAPER 0; INK
7;t: LET vj=vj+1: GOSUB 8100: NEXT n
3500 IF aci1=4 THEN LET int=int+1: GOSUB 3700: GOTO
3730
3510 LET int=int+1: LET e=e-4: LET aci1=0: LET
aci2=0: LET v=v-4
3520 LET c#=STR# int: IF LEN c#>1 THEN PRINT AT
0,13; PAPER 6; INK 0;int
3525 IF LEN c#=1 THEN PRINT AT 0,13; PAPER 6; INK
0;int
3530 IF int=12 THEN GOSUB 3705: GOTO 3900
3532 IF int=6 THEN LET e=23: LET v=22: PRINT AT
1,2; PAPER 2; INK 7;"t12tt11tt10t 9 t8tt 7 "
3535 GOTO 3000
3705 LET kj=a: LET li=r: LET co=28: GOSUB 3015
3710 LET kj=b: LET li=s: GOSUB 3015
3715 LET kj=c: LET li=t: GOSUB 3015
3720 LET kj=d: LET li=u: GOSUB 3015: PAUSE 300:
RETURN
3730 POKE 23607,246
3731 FOR n=1 TO 2: LET x#="T24002bb9E2bEb3E9B"
3732 LET y#="T24004bb9E2bEb3E9B"
3733 LET w#="T24006bb9E2bEb3E9B":tx#,y#,w#: NEXT n
3734 BORDER 5: PAPER 5: INK 0: CLS : PRINT AT
8,3;"SZEP MUNKA VOLT,EZ IGEN!!!"
3735 IF int<rec THEN PRINT AT 12,5;"A MAI NAPON
MEGDONTOTTED A";AT 14,12;"CSUCSOT!!!": LET rec=int
3740 PRINT #1;"MEGEGYSZER ? (I/N)"
3741 FOR n=1 TO 2: LET x#="T17003N1cge3C"
3742 LET y#="T17005N1cge3C"
3743 LET w#="T17007N1cge3C"
3744 tx#,y#,w#: NEXT n
3745 LET d#=INKEY#: IF d# <> "i" AND d# <> "n" THEN
GOTO 3745
3750 IF d#="n" THEN NEW
3755 IF d#="i" THEN GOTO 1000
3900 POKE 23607,246
3901 FOR n=1 TO 4: LET x#="T17003N1cge3C"
3902 LET y#="T17005N1cge3C"
3903 LET w#="T17007N1cge3C"
3904 tx#,y#,w#: NEXT n: PAUSE 150: BORDER 0: PAPER
0: INK 7: CLS : PRINT AT 8,3;"HAT,MOST EZ NEM
SIKERULT..."
3905 GOTO 3740

```


PUZZLE 2004

A játék egyszerű - csak helyére kell tologatni a kockákat, legalább 200 lépés kell a megoldáshoz, gyermekkorom kedvenc játéka volt...



```

1-REM
2-REM *****
10-REM *© NyitraiSOFT 2004*
20-REM *****
30-REM *****
40-REM ***** PUZZLE JÁTÉK *****
50-REM *****
60-REM *****
70-REM *****
80-REM *****
90-REM *****
100-REM *****
110-REM ***** ABC PUZZLE *****
120-REM * Nyitrai László *
130-REM ***** 48K BASIC *****
140-REM ***** @j karakter *****
150-DIM c$(16): DIM y$(1)
160-BORDER 1: PAPER 4: INK 0: CLS
170-GO SUB 860
180-PRINT INK 1; AT 2,8; "j t k g y r o s 2004"; INK 2; AT
3,11; " 48K BASIC"; AT 5,8; INK 3; BRIGHT 1; "ABC puzzle
j t k"
190-LET a$=" ABCDEFGHIJKLMNOP"
200-LET b$=a$(2 TO 4)+" "
210-LET d$="@@U": LET e$=" @ @": LET f$="o u": LET
g$=" "
220-PRINT AT 6,7; INK 1; "Rakd össze ABC sorba!!"
230-PRINT AT 8,10; d$; d$; d$; d$
24-PRINT AT 11,10; d$; d$; d$; d$
25-PRINT AT 14,10; d$; d$; d$; d$

```



```

590-PRINT PAPER 6; INK 1;AT s-1,o-1;d$;AT s,o-
1;e$;AT s+1,o-1;f$
600-PRINT PAPER 6; INK 2;AT s,o;y$
610-REM BEEP .01,(4 AND s>s1)+(7 AND o<o1)+(12 AND
o>o1)
612-RANDOMIZE USR 65055
620-LET k=1: FOR i=6 TO 15 STEP 3: FOR j=11 TO 20
STEP 3: LET c$(k)=SCREEN$(i,j)
630-LET k=k+1: NEXT j: NEXT i
640-LET e=e+1: PRINT AT 1,19; FLASH 1;e; FLASH 0
650-IF b$<>c$ AND a$<>c$ THEN GO TO 520
660-PRINT INK 2;AT 2,10; FLASH 1;"!!!SIKER!!!":
FLASH 0
670-FOR i=11 TO 1 STEP -1: BEEP .1,i: NEXT i: BEEP
.3,i
680-IF e<min THEN LET min=e: PRINT INK 1;AT 3,10;"[K]j
rekord! ";e
690-PRINT #0;"Nyomd az ENTER-t "; FLASH 1;" az [K]j
j[Q]t1khoz!"; FLASH 0
700-IF INKEY#="" THEN GO TO 700
710-GO TO 280
720-BEEP .1,-10: BEEP .2,-20: BEEP .3,-30: PRINT AT
2,10;"Nem siker3lt!": PRINT AT 19,3;"
725-PRINT AT 21,8; INK 1;"NYITRAISOFT 2004"
730-PRINT #0;"Nyomd az ENTER-t"; FLASH 1;" az [K]j
j[Q]t1khoz"; FLASH 0
740-PAUSE 0: PAUSE 0
750-GO TO 290
860-INK 9: PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW -
255,0: DRAW 0,-175: PLOT 1,1: DRAW 253,0: DRAW 0,173:
DRAW -253,0: DRAW 0,-173
870-PLOT 4,4: DRAW 247,0: DRAW 0,167: DRAW -247,0:
DRAW 0,-167
880-RETURN

```

SZÓKERESŐ JÁTÉK

A táblán elrejtett 1-10 szót kell megtalálni, a megtalált szó más színű lesz, segítséget is lehet kérni, a betűk különlegesek..ha meguntad, akkor fel is adhatod.



```
1 REM SZOKERESO JATEK
  10 REM JATEKGYAROS 2004'
  15 GOSUB 5000: GOSUB 5500
  20 BORDER 0: PAPER 0: INK 7: CLS : POKE 23658,8
  22 POKE 23607,246
  90 PRINT AT 9,9;"SZOKERESO": LET a$="  JATEK":
FOR g=1 TO 10: FOR f=1 TO 7: PRINT AT 11,f+11;a$(f):
POKE 23728,11: POKE 23729,f+11: BEEP .05,33:
RANDOMIZE USR 23300: NEXT f: NEXT g
  95 INK 7: PAUSE 200: GOSUB 9000
  96 BORDER 0
  100 REM A SZAVAK BEKERESE
  110 CLS : INPUT "HANY SZO LESZ? (MAX. 10): ";npal:
IF npal>10 OR npal<1 THEN GOTO 110
  115 DIM p$(npal,10): DIM z$(npal,7)
  117 FOR f=1 TO npal
  120 INPUT (f);"- SZO ";": ": LINE a$: IF LEN a$<2
OR LEN a$>10 THEN BEEP .5,-33: GOTO 120
  125 LET p$(f)=a$
  126 IF f>1 THEN FOR g=1 TO f-1: IF p$(f)=p$(g)
THEN INPUT "": PRINT #0;">> EZ A SZO MAR SZEREPELT
<<": PAUSE 100: GOTO 120
  127 IF f>1 THEN NEXT g
  130 NEXT f
  140 GOSUB 7000
  150 INPUT "": PRINT #0: FLASH 1: INK 2:"tttttt";
FLASH 0;" EGY KIS TURELMET "; FLASH 1:"tttttt"
  1000 REM SZAVAK ELTUNTETESE
  1010 DIM x$(17,17,1): RANDOMIZE PEEK 23672
  1020 FOR f=1 TO npal
  1025 POKE 23673,0: GOSUB 8000
  1030 LET dir=INT (RND*8)+1
```

```

1040 LET a$=p$(f): GOSUB 7500: FOR c=1 TO LEN a$:
IF a$(c) <> " " THEN NEXT c
1050 LET len=c-1
1055 IF PEEK 23673 >= 15 THEN BEEP 1,-33: FOR f=7
TO 21: PRINT AT f,0;"tttttttttttttttt": NEXT f: INPUT
"": PRINT #0;" CSEND LEGYEN,ELTEVEESZTETTEM!! ": BEEP
2,33: GOTO 150
1056 LET posx=INT (RND*17)+1: LET posy=INT
(RND*17)+1: IF x$(posx,posy) <> a$(1) AND
x$(posx,posy) <> CHR$ 32 THEN GOTO 1056
1060 GOTO 1000+100*dir
1100 IF posx+len >= 18 THEN GOTO 1055
1105 FOR v=posx TO (len+posx)-1: IF x$(v,posy) <>
CHR$ 32 AND x$(v,posy) <> p$(f)((v+1)-posx) THEN GOTO
1055
1106 NEXT v
1110 LET c=1: FOR g=posx TO (len+posx)-1: LET
x$(g,posy)=a$(c): LET c=c+1: NEXT g
1120 GOTO 1900
1200 IF posx-len <= 0 THEN GOTO 1055
1205 FOR v=posx TO (posx-len)+1 STEP -1: IF
x$(v,posy) <> CHR$ 32 AND x$(v,posy) <>
p$(f)((posx+1)-v) THEN GOTO 1055
1206 NEXT v
1210 LET c=1: FOR g=posx TO (posx-len)+1 STEP -1:
LET x$(g,posy)=a$(c): LET c=c+1: NEXT g
1220 GOTO 1900
1300 IF posy+len >= 18 THEN GOTO 1055
1305 FOR v=posy TO (len+posy)-1: IF x$(posx,v) <>
CHR$ 32 AND x$(posx,v) <> p$(f)((v+1)-posy) THEN GOTO
1055
1306 NEXT v
1310 LET c=1: FOR g=posy TO (len+posy)-1: LET
x$(posx,g)=a$(c): LET c=c+1: NEXT g
1320 GOTO 1900
1400 IF posy-len <= 0 THEN GOTO 1055
1405 FOR v=posy TO (posy-len)+1 STEP -1: IF
x$(posx,v) <> CHR$ 32 AND x$(posx,v) <>
p$(f)((posy+1)-v) THEN GOTO 1055
1406 NEXT v
1410 LET c=1: FOR g=posy TO (posy-len)+1 STEP -1:
LET x$(posx,g)=a$(c): LET c=c+1: NEXT g
1420 GOTO 1900
1500 IF posx+len >= 18 OR posy+len >= 18 THEN GOTO
1055
1505 FOR v=0 TO len-1: IF x$(posx+v,posy+v) <> CHR$
32 AND x$(posx+v,posy+v) <> p$(f)(v+1) THEN GOTO 1055
1506 NEXT v
1510 FOR g=0 TO len-1: LET
x$(posx+g,posy+g)=a$(g+1): NEXT g
1520 GOTO 1900
1600 IF posx-len <= 0 OR posy-len <= 0 THEN GOTO
1055
1605 FOR v=0 TO len-1: IF x$(posx-v,posy-v) <> CHR$
32 AND x$(posx-v,posy-v) <> p$(f)(v+1) THEN GOTO 1055
1606 NEXT v
1610 FOR g=0 TO len-1: LET x$(posx-g,posy-
g)=a$(g+1): NEXT g
1620 GOTO 1900
1700 IF posx+len >= 18 OR posy-len <= 0 THEN GOTO
1055
1705 FOR v=0 TO len-1: IF x$(posx+v,posy-v) <> CHR$
32 AND x$(posx+v,posy-v) <> p$(f)(v+1) THEN GOTO 1055
1706 NEXT v
1710 FOR g=0 TO len-1: LET x$(posx+g,posy-
g)=a$(g+1): NEXT g
1720 GOTO 1900
1800 IF posx-len <= 0 OR posy+len >= 18 THEN GOTO
1055
1805 FOR v=0 TO len-1: IF x$(posx-v,posy+v) <> CHR$
32 AND x$(posx-v,posy+v) <> p$(f)(v+1) THEN GOTO 1055
1806 NEXT v
1810 FOR g=0 TO len-1: LET x$(posx-
g,posy+g)=a$(g+1): NEXT g
1900 LET z$(f)(1 TO 2)=STR$ posx: LET z$(f)(3 TO
4)=STR$ posy: LET z$(f)(5 TO 6)=STR$ len: LET
z$(f)(7)=STR$ dir: NEXT f

```

```

19002 FOR f=5 TO 21: PRINT AT
f,0;"tttttttttttttttttttttttttttt": NEXT f
19005 INPUT "": PRINT #0; FLASH 1; INK 6;"tttttttt";
FLASH 0;" ROGTON VEGZEK "; FLASH 1;"tttttttt"
1910 DIM e$(17,17): FOR f=1 TO 17: FOR g=1 TO 17:
IF CODE X$(f,g)=32 THEN LET X$(f,g)=CHR$(INT
(RND*25)+65)
1920 LET e$(g)(f)=X$(f,g): NEXT g: NEXT f
1930 BEEP 1,33: INPUT "": FOR f=5 TO 21: PRINT AT
f,0;"#####": BEEP .01,33: NEXT f: FOR
f=21 TO 5 STEP -1: PRINT AT f,0;e$(f-4): BEEP .01,33:
NEXT f
1940 INPUT "": PRINT #0;"NYOMD AZ ""F""-T,HA
FELADOD."
2000 REM AZ IRANYITAS
2010 LET posy=9
2020 PRINT AT posy,18;"t"
2030 IF INKEY#=CHR$ 10 THEN BEEP .1,33: PRINT AT
posy,18;" ": LET posy=posy+1: IF posy=npal+9 THEN LET
posy=npal+8
2040 IF INKEY#=CHR$ 11 THEN BEEP .1,33: PRINT AT
posy,18;" ": LET posy=posy-1: IF posy=8 THEN LET
posy=9
2050 IF INKEY#="" THEN GOSUB 6000
2055 IF INKEY#="P" THEN GOSUB 9500
2060 IF INKEY#="F" THEN GOTO 100
2070 GOTO 2020
5000 REM A PROGRAM ESZE
5010 RESTORE 5050
5020 FOR f=23300 TO 23344
5030 READ a: POKE f,a
5040 NEXT f: RETURN
5050 DATA
50 176,92,71,58,177,92,79,62,0,22,4,245,205,30,91,126
,15,119,241,60,21,200,195,16,91,245,120,230,24,246,64
,103,241,132,103,120,230,7,15,15,15,129,111,201
5050 REM SAJAT GRAFIKA
50510 RESTORE 5550
50520 FOR f=USR "a" TO USR "i"+7
50530 READ a: POKE f,a
50540 NEXT f: RETURN
50550 DATA
251,23,47,223,0,0,0,0,123,177,218,239,180,24,176,240,
144,144,144,160,208,176,112,240
50551 DATA
222,141,219,247,45,24,13,15,9,9,9,5,11,13,14,15,15,13
,24,45,247,219,141,222
50552 DATA
0,0,0,0,239,23,11,253,240,176,24,180,239,219,177,123,
126,195,165,129,165,153,195,126
6000 REM OSSZEKEVERTEM MINDENT
6010 LET x1=VAL z$(posy-8)( TO 2): LET y1=VAL
z$(posy-8)(3 TO 4): LET len=VAL z$(posy-8)(5 TO 6):
LET dir=VAL z$(posy-8)(7)
6020 GOTO 6000+100*dir
6100 LET x2=x1+(len-1): LET y2=y1: LET stx=1: LET
sty=1: GOTO 6900
6200 LET x2=x1-(len-1): LET y2=y1: LET stx=-1: LET
sty=1: GOTO 6900
6300 LET x2=x1: LET y2=y1+(len-1): LET stx=1: LET
sty=1: GOTO 6900
6400 LET x2=x1: LET y2=y1-(len-1): LET stx=1: LET
sty=-1: GOTO 6900
6500 LET x2=x1+(len-1): LET y2=y1+(len-1): LET
stx=1: LET sty=1: GOTO 6950
6600 LET x2=x1-(len-1): LET y2=y1-(len-1): LET
stx=-1: LET sty=-1: GOTO 6950
6700 LET x2=x1+(len-1): LET y2=y1-(len-1): LET
stx=1: LET sty=-1: GOTO 6950
6800 LET x2=x1-(len-1): LET y2=y1+(len-1): LET
stx=-1: LET sty=1: GOTO 6950
6900 RANDOMIZE PEEK 23672+256*PEEK 23673: LET
bright=INT (RND*2): LET ink=INT (RND*7)+1: FOR f=x1
TO x2 STEP stx: FOR g=y1 TO y2 STEP sty: LET col=f-1:
LET fil=g+4: PRINT AT fil,col; INVERSE 1; OVER 1; INK
ink; BRIGHT bright;" ": POKE 23728,fil: POKE

```

```

23729,col: FOR v=1 TO 8: RANDOMIZE USR 23300: BEEP
.01,33: NEXT v: NEXT g: NEXT f: RETURN
6950 RANDOMIZE PEEK 23672+256*PEEK 23673: LET
bright=INT (RND*2): LET ink=INT (RND*7)+1: FOR f=0 TO
len-1: LET col=((x1+f)-1 AND stx=1)+((x1-f)-1 AND
stx=-1): LET fil=(y1+f+4 AND sty=1)+(y1-f+4 AND sty=-
1): PRINT AT fil,col: INVERSE 1: OVER 1: INK ink;
BRIGHT bright; "A: POKE 23728,fil: POKE 23729,col:
FOR v=1 TO 8: RANDOMIZE USR 23300: BEEP .01,33: NEXT
v: NEXT f: RETURN
7000 REM KEPERNYO
7005 LET t$=" SZOKERESO JATEK
NYITRAISOFT JATEKGYAROS 2004' MAJUS"
7010 PRINT INK 2;AT 1,6;t$
7020 PRINT INK 5;AT
0,0;"tttttttttttttttttttttttttttttttttttttttttttt"; INK 5;AT
4,0;"tttttttttttttttttttttttttttttttttttttttttttt"; INK 5;AT
21,17;"tttttttttttttttttttttttttttttttttttttttttttt";
7030 FOR f=1 TO 3: PRINT INK 5;AT f,0;"t"; INK 5;AT
f,31;"t": NEXT f
7040 PRINT INK 5;AT 4,31;"t": FOR f=5 TO 20: PRINT
INK 5;AT f,17;"t"; INK 5;AT f,31;"t": NEXT f: PRINT
INK 5;AT 4,17;"t"
7050 REM OVER 1: FOR f=170 TO 140 STEP -2: INK
1: PLOT 5,f: DRAW 245,0: NEXT f: OVER 0
7060 INK 4: FOR f=5 TO 21 STEP 2: PRINT AT
f,0;"tttttttttttttttttttttttttttttttttttttttttttt": BEEP .1,33: NEXT f
7065 FOR f=20 TO 5 STEP -2: PRINT INK 4;AT
f,0;"tttttttttttttttttttttttttttttttttttttttttttt": BEEP .1,33: NEXT f
7070 PRINT INK 3;AT 7,18;"KERESEENDOK:"; INK 2;AT
7,18: OVER 1; "-----"
7080 FOR f=1 TO npal: LET a#=p$(f): GOSUB 7500: FOR
g=1 TO v: PRINT AT 8+f,18+g;a$(g): POKE 23728,8+f:
POKE 23729,18+g: BEEP .01,0: RANDOMIZE USR 23300:
NEXT g: NEXT f
7090 RETURN
7500 FOR v=1 TO 10: IF CODE a$(v) <> 32 THEN NEXT v
7505 LET v=v-1
7510 LET a#=a$( TO v)
7520 RETURN
7777 PRINT OVER 0;AT 21,0;a$,: FOR v=1 TO 15: FOR
g=1 TO 15: PRINT AT v,g;x$(g,v): PAUSE 0: NEXT g:
NEXT v: RETURN
8000 REM BEOLVASAS
8020 RESTORE 8500: FOR v=0 TO npal-f: READ c$: NEXT
v
8030 INVERSE 1: PRINT INK 2;AT 11,2;" MAR CSAK"; ("
" AND npal-f <> 1); (" " AND npal-f=1);AT 12,2;"
";c$;AT 13,2;" SZO VAN"; (" " AND npal-f <> 1); ("
" AND npal-f=1);AT 14,2;" HATRA
8035 INVERSE 0
8040 RETURN
8499 REM SZAVAK SZAMA
8500 DATA "AZ UTOLSO ","EGY ","KETTO
","HAROM ","NEGY ","OT ","HAT
","HET ","NYOLC ","KILENC "
9000 REM MAGYARAZAT
9010 LET A$="EBBEN A JATEKBAN 10 HOSSZU ES MAX.10
DARAB SZOT KELL AZ ELE- JEN BEIRNUNK,MELYET A
SPECTRUM EGY 17X17-ES SZOVEGMATRIXBAN HELYEZ,REJT
EL,MELYET MEG KELL TALALNUNK.HA EZ SIKERULT,AKKOR A
'0' LENYOMASAVAL MEG IS JE- LENITI NEKUNK,MAJD A
KURZORRAL A KOVETKEZO SZORA LEPHETUNK ES A KERESTET
TOVABB FOLYTATHAT- JUK.A 'P'-VEL NYOMTATORA KULD-
HETJUK A TABLAT,ES AZ 'F'-LE- NYOMASAVAL
FELADHATJUK,VAGY UJ JATEKOT KERHETUNK."
9020 INK 5: CLS : PRINT AT 0,0: INVERSE 1;"SZOREJTO
JATEK NYITRAISOFT 2004' "
9030 LET border=0: FOR f=1 TO LEN a$: PRINT a$(f);:
BEEP (.005 AND CODE a$(f) <> 32)+(0 AND CODE
a$(f)=32),33
9032 IF A$(F) <> CHR$ 32 THEN BORDER border
9035 IF LEN INKEY# THEN RETURN
9036 LET border=NOT border: NEXT f
9040 PRINT #0;" NYOMD LE AZ ENTER-T": PAUSE
0
9050 RETURN

```



```

9500 REM NYOMTATAS
9510 INPUT "": PRINT #0;"          NYOMTATOM ..."
9520 FOR f=1 TO npal: LPRINT e$(f);" ";f;"-
";p$(f): NEXT f
9530 FOR f=f TO 17: LPRINT e$(f): NEXT f
9540 GOSUB 5000: INPUT "": PRINT #0;"NYOMD AZ
""F""-T,AZ UJ JATEKHOZ"
9550 RETURN

```

ÚJTÉGLA-FALTÖRŐ

Készülvén a 20 éves évfordulóra elővettem 1983-ban kapott (első) Spectrum demonstrációs kazettát,- valószínűleg nektek is ez volt az első kazettátok - melyet először kipróbáltatok,és azon lévő Psion cég 1982-es WALL nevű programját lejátszva,úgy gondoltam,hogy az évfordulót úgy kéne nekem megünnepelni,hogy ezt a 48k-ra készült programot, a mai igényeknek megfelelően átalakítom 128K-s programmá,egy kicsit felgyorsítva és többszólamú alafestő zenét kreálva hozzá.Elővettem a ZX Spectrum 128K-2-es gépemet,és ezzel készítettem el az új változatot.Az aláfestő zenét némi próbálkozás után elkészítettem,ezt volt a dolog neheze,persze a régi labdát és ütőt is átalakítottam kissé.A játék menürészét meg úgy alakítottam,mintha a 128K-s gép bejelentkező képe lenne.

A régi WALL és az új UJTÉGLA programokat a levelemhez mellékelem,érdeemes egymás után kipróbálni.

Játékgyáros (e nevémet egyébként Edy találta ki,szívesen használom !!)



```

1 REM TEGLATORES
2 REM (c) NYITRAISOFT 2002
3 REM KESZITETTE: JATEKGYAROS
4 REM A PROGRAM EREDETIJE A ZX SPECTRUM
48K DEMONSTRACIOS KAZETTAJAN A WALL
PROGRAM !!
10 POKE 64062,255: POKE 64001,25: LET tt=-1: LET
f$=CHR$(PEEK 30000): LET g$=CHR$(PEEK 30001): LET
t$=CHR$(PEEK 30002)
20 LET tt=tt+1: LET t=0: LET p=1: BORDER 7: INK
0: PAPER 7
30 FOR m=1 TO 4 STEP 2
32 FOR n=0 TO 30 STEP 2
34 PRINT AT m+3,n; PAPER m+1; INK 0;"tt"; PRINT
AT m+4,n; PAPER m+2; INK 0;"tt"
36 NEXT n: NEXT m
38 FOR n=0 TO 30 STEP 2
40 PRINT AT m+3,n; PAPER 6; INK 0;"tt": NEXT n
46 LET u=0: LET v=0
47 LET w=0
48 LET a=14: LET t=0
50 FOR r=1 TO 6
52 LET m=10: LET n=8+INT(RND*14)
54 LET g=200: LET p=0: LET a=13
56 PRINT AT 21,0;" ttt
"
65 GOSUB g: PRINT AT u,v;" "
70 PRINT AT m,n;"t": LET u=m: RANDOMIZE: LET
v=n: RANDOMIZE USR 64000
74 IF m=20 THEN PRINT AT m,n;"t"
80 PAUSE 1: LET a$=INKEY$: IF a$ =f$ THEN GOSUB
224
86 IF a$ =g$ THEN GOSUB 234
88 IF a$ =t$ THEN GOTO 410
90 GOTO 60
100 IF m>20 THEN GOTO 240
101 IF m<20 THEN GOTO 106
102 IF t >= 558 THEN GOTO 20
103 LET p=0: LET w=0: IF n=a+1 OR n=a+2 THEN LET
g=120: GOTO 120
104 IF n=a THEN LET g=180: GOTO 190
105 IF n=a-1 THEN LET g=140: GOTO 140
106 IF n>30 THEN LET g=160: GOTO 160
110 LET m=m+1: LET n=n+1
112 LET c=ATTR(m,n): IF c <> 56 THEN GOSUB 250:
IF p=0 OR w=1 THEN LET g=120
114 RETURN
120 IF n>30 THEN LET g=140: GOTO 140
125 IF m<1 THEN LET w=1: LET g=100: GOTO 100
130 LET m=m-1: LET n=n+1
132 LET c=ATTR(m,n): IF c <> 56 THEN GOSUB 250:
LET p=1: LET g=100+40*w
134 RETURN
140 IF m<1 THEN LET w=1: LET g=160: GOTO 160
145 IF n<1 THEN LET g=120: GOTO 120
150 LET m=m-1: LET n=n-1
152 LET c=ATTR(m,n): IF c <> 56 THEN GOSUB 250:
LET p=1: LET g=160-40*w
154 RETURN
160 IF m>20 THEN GOTO 240
161 IF m<20 THEN GOTO 166
162 IF t >= 558 THEN GOTO 20
163 LET p=0: LET w=0: IF n=a+3 THEN LET g=120:
GOTO 120
164 IF n=a+2 THEN LET g=180: GOTO 180
165 IF n=a OR n=a+1 THEN LET g=140: GOTO 140
166 IF n<1 THEN LET g=100: GOTO 100
170 LET m=m+1: LET n=n-1
172 LET c=ATTR(m,n): IF c <> 56 THEN GOSUB 250:
IF p=0 OR w=1 THEN LET g=140
174 RETURN
180 IF m<1 THEN LET g=200: GOTO 212
190 LET m=m-1: LET c=ATTR(m,n): IF c <> 56 THEN
GOSUB 250: LET g=200
195 RETURN
200 IF m>20 THEN GOTO 240

```

```

0000 IF m<20 THEN GOTO 212
0001 IF t >= 558 THEN GOTO 20
0002 LET p=0: LET w=0: IF n=a+2 THEN LET g=120:
GOTO 120
0003 IF n=a+1 THEN LET g=20*(6+INT (RND*2)): GOTO g
0004 IF n=a THEN LET g=140: GOTO 140
0005 LET m=m+1: RETURN
0006 IF a<1 THEN RETURN
0007 LET a=a-1: PRINT AT 21,a;"ttt": RETURN
0008 IF a<2 THEN GOTO 220
0009 LET a=a-2: PRINT AT 21,a;"ttt ": RETURN
0010 IF a>20 THEN RETURN
0011 LET a=a+1: PRINT AT 21,a-1;" ttt": RETURN
0012 IF a>27 THEN GOTO 230
0013 LET a=a+2: PRINT AT 21,a-2;" ttt": RETURN
0014 BEEP 1,6: PRINT #1;"t";
0015 NEXT r
0016 BORDER 7: PAPER 7: INK 0: CLS
0017 LET x$="ELERT PONTSZAM " +STR$ t: PRINT AT
10,(32-LEN x$)/2;x$
0018 LET
a$="T100N3ggagdec6ddd3cdeagdg9g3gagdec5dd3edcage9dH":
LET b$="02T100U10N3"+a$ (7 TO ): LET
c$="03T240((1ce))(1gb))":ta$,b$,c$: PAUSE 70: CLS :
GOTO 410
0019 LET t=t+10-c/8
0020 LET b=ABS (m-n)
0021 LET y=INT (b/2)*2
0022 IF b=y AND n<31 THEN PRINT AT m,n+1;" "
0023 IF b <> y AND n>0 THEN PRINT AT m,n-1;" "
0024 RETURN
3000 REM itt a kezdes
3001 BORDER 7: PAPER 7: INK 7: CLS
3002 POKE 23606,160: POKE 23607,166: POKE 64001,165
3003 FOR m=1 TO 4 STEP 2
3004 FOR n=0 TO 30 STEP 2
3005 PRINT AT m,n: PAPER m+1: INK 0;"tt": PRINT AT
m+1,n: PAPER m+2: INK 0;"tt"
3006 NEXT n: NEXT m
4000 PRINT INK 0;AT 1,5;"a SPECTRUM 20 eyes!!!"
4001 POKE 30000,54: POKE 30001,55: POKE 30002,48:
POKE 24456,249: LET i1=7: LET i2=7: LET i3=2: LET
i4=7
4002 POKE 64062,220: PRINT AT 8,8: PAPER 2: INK
i1;"2 IRANYITAS ";AT 7,8: PAPER 1: INK i2;"1
JOYSTICK II ";AT 9,8: PAPER 3: INK 7;"3 ZENE"; INK
i4;" KIKAPCS ";AT 10,8: PAPER 4: INK 7;"4 ZENE"; INK
i3;" BEKAPCS ";AT 11,8: PAPER 5: INK 7;"5 UJRA-
INDITAS ";AT 13,8: PAPER 7: INK 0;"0 JATEK INDITAS"
4003 INK 0: PRINT AT 4,7;"JATEK 128K MENUJE:"
4004 PRINT AT 19,6;"(c) 1982 PSION SOFTWARE";AT
20,6;"(c) 2002 JATEKGYAROS"
4005 PRINT INK 2;AT 17,3;"NYOMJ EGY GOMBOT (0-5) !"
4006 RANDOMIZE USR 64000
4007 IF INKEY$="1" THEN POKE 30000,49: POKE
30001,50: POKE 30002,53: LET i1=7: LET i2=4: GOTO 420
4008 IF INKEY$="2" THEN LET i1=4: LET i2=7: GOTO
6000
4009 IF INKEY$="3" THEN POKE 24456,234: LET i4=1:
LET i3=7: GOTO 420
4010 IF INKEY$="4" THEN POKE 24456,249: LET i3=2:
LET i4=7: GOTO 420
4011 IF INKEY$="5" THEN RANDOMIZE USR 4700
4012 IF INKEY$="0" THEN GOTO 630
4013 GOTO 470
6300 RUN
6000 CLS : INK 0
6001 PRINT AT 10,2;"MOZGAS BALRA:": PAUSE 0: PRINT
AT 10,18;INKEY$: POKE 30000,CODE (INKEY$)
6002 PRINT AT 12,2;"MOZGATAS JOBBRA:": PAUSE 0:
PRINT AT 12,18;INKEY$: POKE 30001,CODE (INKEY$)
6003 PRINT AT 14,2;"JATEK KILEPES:": PAUSE 0: PRINT
AT 14,18;INKEY$: POKE 30002,CODE (INKEY$)
6004 PAUSE 70: CLS : GOTO 415
9999 SAVE "TEGLA128+2" LINE 300

```

Magyar trikolor ZÁSZLÓ

Egy kis játék a 128K betöltőjével, egymásra hívott képek mozgásba hozzák a magyar zászlót, mintha a szél fújná, úgy lobog..

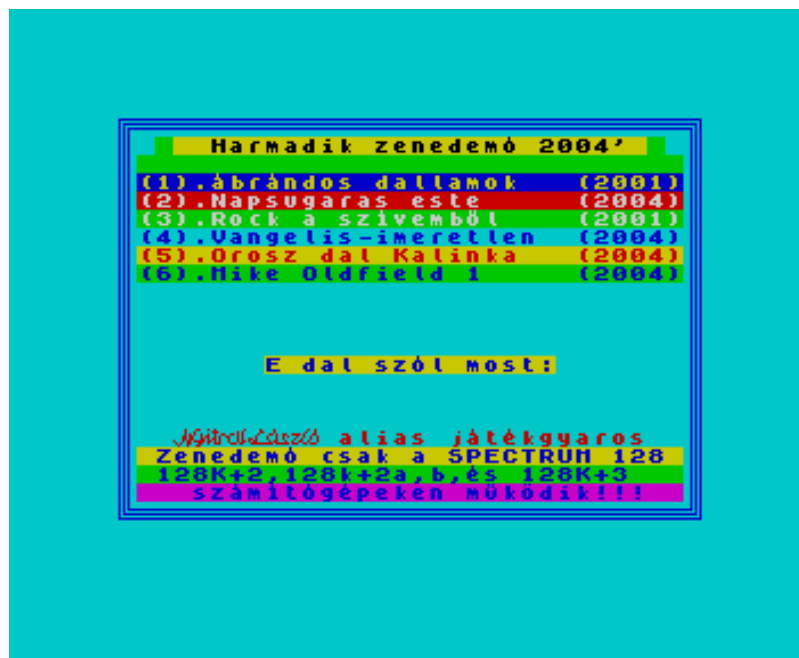


```
1 rem magyar zaszlo
3 rem jatekgyaros 2005-08
10 PAPER 7: BORDER 2: INK 0: CLS
20 FOR n=65 TO 120
30 LET i=2: LET c=5*COS (n/40*PI)
40 PLOT INK i;n,87+c: DRAW INK i;0,33
50 PLOT INK 4;n,56+c: DRAW INK 4;0,-33
60 NEXT n
70 FOR n=100 TO 180: LET c=5*COS (n/40*PI)
80 PLOT INK i;n,87+c: DRAW INK i;0,33
90 PLOT INK 4;n,56+c: DRAW INK 4;0,-33
100 NEXT n
110 FOR v=5 TO 21: PRINT AT v,7;"" : NEXT v
140 SAVE !"zaszlo"SCREEN#
141 CLS
145 FOR n=65 TO 120
147 LET i=2: LET c=COS (n/40*PI)
148 PLOT INK i;n,87+c: DRAW INK i;0,33
149 PLOT INK 4;n,56+c: DRAW INK 4;0,-33
151 NEXT n
153 FOR n=100 TO 180: LET c=COS (n/40*PI)
155 PLOT INK i;n,87+c: DRAW INK i;0,33
157 PLOT INK 4;n,56+c: DRAW INK 4;0,-33
159 NEXT n
161 FOR v=5 TO 21: PRINT AT v,7;"" : NEXT v
170 SAVE !"zaszlo2"SCREEN#
180 LOAD !"zaszlo"SCREEN#
185 LOAD !"zaszlo2"SCREEN#
190 LET a#=INKEY#: IF a#<>"" THEN CLS : GO TO 210
2000 0 TO 180
2010 PRINT #1;"Maradek hely a memoriaban ";65535-USR
7962; " bajt.": PAUSE 0: INPUT PI
2020 AVE "zaszlo"LINE 1
230 STOP
```

HARMADIK ZENEDEMŐ

A harmadik zenedemő:

1. Ábrándos dallamok
2. Napsugaras esete
3. Rock a szívemből
4. Vangelis
5. Kalinka
6. Mike Oldfield



```
1 REM harmadik zenedemő
2 REM készítette Nyitrai László- alias
   játékgyaros 2004'
10 INK 1: PLOT 0,0: DRAW 255,0: DRAW 0,175: DRAW
-255,0: DRAW 0,-175
12 PLOT 2,2: DRAW 251,0: DRAW 0,171: DRAW -251,0:
DRAW 0,-171
13 PLOT 4,4: DRAW 247,0: DRAW 0,167: DRAW -247,0:
DRAW 0,-167
14 PAPER 5: BORDER 5: LET a=3: LET b=1: LET d=0:
LET e=6: PRINT AT a,b: PAPER 1; INK 6;"(1).Ábrándos
dallamok (2001)": PRINT AT a+1,b: PAPER 2; INK
7;"(2).Napsugaras esete (2004)": PRINT AT a+2,b;
PAPER 4; INK 7;"(3).Rock a szívemből (2001)":
PRINT AT 1,2; INK 0; PAPER 4;" "; PAPER 6;" Harmadik
zenedemő 2004' "; PAPER 4;" ": PRINT AT 2,1; PAPER
4;"
15 PRINT AT a+3,b; PAPER 5; INK 1;"(4).Vangelis-
imeretlen (2004)": PRINT AT a+4,b; PAPER 6; INK
2;"(5).Orosz dal Kalinka (2004)": PRINT AT a+5,b;
PAPER 4; INK 1;"(6).Mike Oldfield 1 (2004)"
20 PRINT AT 17,3; INK 2;"tttttttt alias
játékgyaros": PRINT AT 18,1; PAPER 6;" Zenedemő csak
a SPECTRUM 128 "; AT 19,1; PAPER 4;"
128K+2,128k+2a,b,ts 128K+3 "; PAPER 3; AT 20,1;"
számítógépeken működik!!! "
30 PRINT AT 13,8; PAPER 6; INK 1;"E dal szól
most:"
40 IF INKEY$="1" THEN PRINT AT 15,8; INK 0; PAPER
5;"Ábrándos dallamok": GOTO 100
50 IF INKEY$="2" THEN PRINT AT 15,8; INK 0; PAPER
5;"Napsugaras esete": GOTO 180
```

```
5; " 60 IF INKEY#="3" THEN PRINT AT 15,8; INK 0; PAPER
5; "Rock a sztvembtl": GOTO 320
63 IF INKEY#="4" THEN PRINT AT 15,4; INK 0; PAPER
5; "Vangelis ismeretlen ember": GOTO 600
65 IF INKEY#="5" THEN PRINT AT 15,8; INK 0; PAPER
5; "Kalinka": GOTO 700
67 IF INKEY#="6" THEN PRINT AT 15,8; INK 0; PAPER
5; "Mike Oldfield 1": GOTO 800
75 LET d=d+1: IF d=7 THEN LET d=0
80 LET e=e-1: IF e=-1 THEN LET e=6
90 GOTO 30
100 LET t#="T136"
110 LET a#=#t#+ "U14 04 5g5G3FEDCDDDFED#ba ((04
5g5G3FEDCDDDFED#ba)) (04 6g5G3FEDCDDDFED#ba)04
5g5G3FEDCDDDFED#b5a (6g6f5g6a6g5a5g5G3FEDCDDDFED#b5a)5g5G
3FEDCDDDFED#b5a:"
120 LET b#=#t#+ "((U0 5&U13 b6a5b5b3b5aa))04
5g3Db6C5f3fCa#bf#f04 6g3Db6C5f3fCa#bf#f04
5g3Db6C5f3fCa#bf#f04 6g3Db6C5f3fCa#bf#f04
5g6a5b5b3b5aa)5g6a5b5b3b5aa:"
130 LET c#=#t#+ "((U0 5&U13 d6c5d5d3d5cc))U15 (05
5d5D6C5D3D5E7F7E5D6C5C3agffe fg)(04 9#bC
5g5d6c5d5d3d5ccc)5g5d6c5d5d3d5ccc"
140 LET d#=#t#+ "(05
5d5D6C5D3D5E7F7E5D6C5C3agffe fg)(04 9#bC
5g5d6c5d5d3d5ccc)5g5d6c5d5d3d5ccc04
9#bC5g5d6c5d5d3d5ccc04 9#bC7D UX5000W0 M14 08 3c07
1c06 3c05 1c04 3c03 5c:"
150 LET e#=#t#+ "04 5g3Db6C5f3fCa#bf#f04
6g3Db6C5f3fCa#bf#f04 6g3Db6C5f3fCa#bf#f04
5g6a5b5b3b5aa)5g6a5b5b3b5aa05 9#e f04 5g6a5b5b3b5aa
05 9#e f7g"
160 LET f#=#t#+ "((04 5g5G3FEDCDDDFED#b5a))
(6g6f5g6a6g5a5g5G3FEDCDDDFED#b5a)5g5G3FEDCDDDFED#b5a6g6f
5g6a6g5a5g5G3FEDCDDDFED#b5a 6g6f5g6a6g5a7g"
170 ta$,b$,c$:td$,e$,f$: CLS : GOTO 10
180 LET t#="T240"
190 LET a#=#t#+ "(U14 03
((5aEG)aE(eEG)gE))((5aEG)aE(eEG)gE)) (U13 05
(5egb)eg(#fgb)#fg (gbE)gb(abE)ab)"
200 LET b#=#t#+ "U12 05
(((10Cae)))(((10bge))) (((10Cae))) (((10bge)))
(((10Cae))) (((10bge))) (((10Cae))) (((10bge)))
(((10Cae))) (((10bge))) (((10Cae))) (((10bge)))
(((10Cae))) (((10bge))) (((10Cae))) (((10bge)))
(((10Cae))) (((10bge))) (((10Cae))) (((10bge)))
U13 05(((10Cge)))
(((10b#fd)))(((10Ebe))) (((10ba#f))) (((10gec)))
) (((10b#fd))) (((10Ebe))) (((10ba#f)))"
210 LET c#=#t#+ "U11 (((10
ECa))) (((10Dbg))) (((04 10 ECa))) (((04
10Dbg))) (((05 10ECa))) (((10Dbg))) (((04
10ECa))) (((10Dbg))) (U14 UX4000W2 05
9_9_9_8E3DC9_9_9_5a5aCD7EG6ED5C9_8D3ED9_9_9_9C) (U13
03(5Ce)g(Cg(D#fa)Da(Egb)Eg(#FaD)#FD)"
220 LET d#=#t#+ "U11 ((3egbEEbge)(d#faDDa#fd)) (05
U13 9_9_8E5G#FDba9_9_7_3g3ababDbDbDbD9_9_8E5DEGAB06
9_9_9_9E)UX3000W0_05 9_7E
7EED5ba7g9_7E7G5#FG#FD9a9_7E7EED5ba7gega9_9_5_5b"
230 LET e#=#t#+ "U12 05
(((10GEb)) ((10#FDb)) ((10GEb)) ((10BGE)) ((10A#FD)) ((10
#FDa)) ((10EDa)) ((10Da#f)))
(((10GEb)) ((10#FDb)) ((10GEb)) ((10BGE)) ((10A#FD)) ((10
#FDa)) ((10EDa)) ((10Da#f))) (((10GEb)) ((10#FDb)) ((10GE
b)) ((10BGE)) ((10A#FD)) ((10#FDa)) ((10EDa)) ((10Da#f)))
04 U12
(((10ECg))) (((10#FDa)) ((10GDa)) ((10#FDa)) (((10ECg
))) (((10#FDa))) (((10ECg))) (((10#FDa)) ((10GDa)) (
(10#FDa)) (((10ECg))) ((10B#F#D)) ((10B#F#D)) ((10B#F
#D))"
240 LET f#=#t#+ "UX5000W002
((5EBbB)(DAaA)) ((5EBbB)(DAaA)) ((5EBbB)(DAaA)) (U13
03 SCE5&SCE5&CED#FS&DG#FDa) SCE5&SCE5&CED#FS&DG#FDa
SCE5&SCEGA9_5_5_5_5_5B"
```



```

650 ta#,b#,c#
660 td#,e#,f#: CLS : GOTO 10
700 REM Kalinka
710 LET
a$="05U14T140N6#D5#C3#ab5#C3#ab5#C3b#a5#gT145N3#D#D4#
C1b3#ab5#C3#ab5#C3b#a5#g"
715 LET
b$="05U14T140N6#g(5#a3g#g)5#a3gg5#gT145N3bb4#a1#g3g#
g5#a3g#g5#a3gg5#g"
720 LET
c$="03U13T140N6#g(3#D#A#D#A)3#D#A#D#A#GBT145b#G#D#A#D
#A(3#D#A#D#A)#GB"
725 LET
d$="05U14T150N5#D5#C3#ab5#C3#ab5#C3b#a5#gT155N3#D#D4#
C1b3#ab5#C3#ab5#C3b#a5&6#GN7#F"
730 LET
e$="05U14T150N5#g(5#a3g#g)5#a3gg5#gT155N3bb4#a1#g3g#
g5#a3g#g5#a3gg5#g6&U15N6#CU13N3E"
735 LET
f$="03U13T150N5#g(5#D3#D#A)5#D3#D#A#GBT155#g b(#D#A)5#
D3#D#A3#D#A#D#D5#g6&7#f"
740 LET
g$="05U14T145N3#D#F3#D1#D#C5b#f3#D#FE1#D#C5b#f#g3#g#a
#Cb#a#g5#f#f7#f7#F"
745 LET
h$="05U14T145N5b#C3b#D#C1b#a5#f1#c#%c#c#c5#d#c7#e#g#f
g#f"
750 LET
i$="03U14T145N5b3#f1#g#a5b#D3b1#Cb#a#f#g#a5b#D7E#C5#f
#99a"
755 LET
j$="05U14T160N(3#D#F#C#D5b#f)5#g3#g#a c b#a#g5#FE7#D"
760 LET
k$="05U14T160N(3b#D#a#a5#f#d)5e3ee5#g#g#a#a7#a"
765 LET
l$="03U14T160N(5b3#F#F1B#A#G#A5#F)5E3EE5#C#C#C7#D"
770 LET
m$="05U14T150N5#D5#C3#ab5#C3#ab5#C3b#a5#gT155N3#D6#D4
#C1b3#ab5#C3#ab5#C3b#a5#gU15N3#G&H"
775 LET
n$="05U14T150N5#g&5#a3g#g)5#a3gg5#gT155N3bb4#a1#g3g#
g5#a3g#g5#a3gg5#gU15N3#b&H"
780 LET
o$="03U13T150N5#g(5#D3#D#A)5#D3#D#A#GBT155#g b(#D#A)5#
D3#D#A3#D#A#D#D5#gU15N3#G&H"
785 ta#,b#,c#
786 td#,e#,f#
787 tg#,h#,i#
788 tj#,k#,l#
789 ta#,b#,c#
790 tm#,n#,o#
795 CLS : GOTO 10
800 REM Mike Oldfield Napsugtr
805 LET
a$="T20005U12((3eg8C)3eg5CDDEE&&3CD5EGECagg3ef5gC3CaC
D5C&&)3&&04N7E5DCaggegedc5d&&5e3dc5dfgg&&&7G5FEFEDCag
CDG&&GRA05N5CGEDCaage3agCD5C&&H"
810 LET b$="5&M2104U9N1c&&&U13N1C&3&)"
815 LET
c$="5&03U12(7cfcccgc cccfcccgcc)cfcccgggcfcccgcc fccggc
ccfc"
820 LET
d$="T200N5&05U12N7D#FDa5Cbab7b&D#FAB06N3CEDa bDCbab5g&
H"
825 LET e$="5&03U12((7d))((g))((d))gdgg"
830 LET
f$="T20006N5&U12N3gecd5fdedc&3gecd5fdeag&05N3gaCD5FDE
DC&gC3CaCD5C7&H"
835 LET g$="5&03U12(7cfccc f5cg7c)"
840 LET h$="T200N5&06U10N9_7_7gUX500W4N9_7_7gH"
850 LET i$="04U11N5c9_7_7c"
895 ta#,b#,c#
896 ta#,b#,c#
897 td#,b#,e#
898 ta#,b#,c#
899 td#,b#,e#:tf#,b#,g#:ta#,b#,c#:th#,b#,i#: CLS :
GOTO 10

```

FÜGGELÉK:

SZÖVEGES KALANDJÁTÉK

Szöveges kalandjátékok készítésének gyakorlati problémái.

1. TARTALOMJEGYZÉK

1. TARTALOMJEGYZÉK
2. ÖSSZEFOGLALÓ
3. KIDOLGOZÁS
 - 3.1. A kalandjáték-készítés általános tudnivalói
 - 3.1.1. A kalandjáték fogalmának tisztázása
 - 3.1.2. Kalandjáték és mitológia
 - 3.1.3. A számítógép beszélni tanul
 - 3.1.4. A szavaktól a mondatok felé
 - 3.1.5. Helyiségek összefüggő labirintusa
 - 3.1.6. Barangolás a térképen
 - 3.1.7. Lakberendezővé változunk
 - 3.1.8. Hogyan találjunk meg valamit?
 - 3.1.9. "Sokasodjatok és növekedjetek!"
 - 3.1.10. Az idő kerekéhez kötve
 - 3.1.11. Néhány jótanács és további lehetőségek
4. IRODALOM

2. ÖSSZEFOGLALÓ

Az először a kalandjáték fogalmát tárgyaljuk, illetve általánosságban beszélünk, annak megvalósítási formáiról és lépéseiről. Főképpen a hagyományos szöveges kalandjátékról esik szó benne, külön koncentráva a magyar nyelvű szövegértés nehézségeire (főnevek és igék ragozása, kötetlen szórend, ékezetes betűk stb.).

Játékokkal foglalkozni nem feltétlenül tartozik a lenézendő vagy megvetendő tevékenységek közé: minden élőlény egészséges állapota, ha játszik, és különösen pedig a legnagyobb matematikusok mindig is rendkívül játékos

elmék voltak. Itt van pl. mindjárt Neumann János; vagy a John Conway angol matematikus által kiagyalt Életjáték - sejtautomata, mely kiált a számítógépes megvalósítás után. Állítom, hogy a kalandjáték is ugyanebbe a kategóriába tartozik!

3. KIDOLGOZÁS

3.1. A kalandjáték-készítés általános tudnivalói

3.1.1. A kalandjáték fogalmának tisztázása

Az isteni szikra két egyetemista (bizonyos Crowther és Woods nevezetű úriemberek) fejéből pattant ki 1977-ben, amikor is előálltak PDP-10 számítógépre készített - és akkoriban még teljesen eredetinek számítójátékprogramjukkal, melynek az "Adventure" (magyarul: "Kaland") címet adták.

Egy kígyókkal, manókkal és más mesebeli lényekkel benépesített földalatti labirintusban elszórt kincseket kellett a játékosnak megtalálnia. Maga a program a korabeli diákság kedvenc szórakozásává lépett elő; a ráakasztott

elnevezés pedig - az azóta eltelt két évtized során - olyan fogalommá terebélyesedett, melyről ma már szinte mindenkinek más-más dolgok jutnak eszébe. Számítógépjátékok tízezrei viselik büszkén vagy kevésbé büszkén ezt a besorolást, de ha némelyiket közülük egymás mellé raknánk, hát bizony meg sem fordulna a fejünkben a gondolat, hogy ezeknek bármi közülük is lehet egymáshoz.

Az egyik képernyőn egy kicsinyített színes csatateret látunk mintegy madártávlatból, vezényszóra ide-oda rohangáló figurákkal tarkítva; a másikat egy térhatású, penészes falú börtön hatalmas sötét alagútja tölti be; a harmadikon mondatokba és bekezdésekbe tördelt, sűrűn szedett apróbetűs szöveg olvasható. Pedig valamennyi ugyanarról a közös tőről fakad: amiről ebben az írásban szó lesz, az mindezek östípusa, mely mindazonáltal nem pusztult ki és nem tűnt el teljesen ma sem, csupáncsak erősen háttérbe szorította őt ezer ágon fejlődő leszármazottainak hihetetlen burjánzása - pontos meghatározás szerint ez az ún. hagyományos szöveges kalandjáték. A "kalandjáték" szót tehát mostantól kezdve ebben az értelemben használom. Még ha másfajta programot szándékozik készíteni valaki, akkor is erről az alapról kell elindulnia; s noha főképpen ezekről lesz szó, természetesen a téma alapelvei mindenfajta grafikus játékra is vonatkoznak - elvégre a rajzos külső is csak egy burok, amit ha lehántunk róla, a program belsejében hasonlóan kell nyilvántartanunk a dolgokat. (S minekutána egy színvonalas szöveges játékot megírni némileg nehezebb programozói feladat, mint egy grafikus, ez egyúttal jó iskolát is jelent a későbbiek számára.)

Mihelyt egy ilyen programot elindítunk, rendszerint valamilyen különös helyszínen, egy fantáziavilág közepén találjuk magunkat. Kapunk egy bekezdésrevaló leírást vagy valamilyen képet tartózkodási helyünk közvetlen környezetéről - esetleg némi magyarázatot is arról, miképpen kerültünk oda -, és elindulunk fölfedezni világunk ismeretlen titkait. A klasszikus modellben a játék egyfajta párbeszédes üzemmódban működik: mi valamit mondunk a számítógépnek (kérjük vagy utasítjuk valaminek a megtételére), ő pedig üzenetek útján értesít bennünket lépéseink eredményéről. De cselekvéstől függetlenül is történhetnek események. Maga a játék lényegében egy kitalált világ életének szimulációja: ez a világ, hogy könnyebben ábrázolható legyen, határozott, különálló helyiségekre oszlik. Egy-egy ilyen helyszín lehet pl. egy erdei tisztás, egy lépcsőforduló, egy szoba egy házban vagy egy tó vizén ringatózó csónak - általánosságban a terep bármely jellegzetes pontja, mely a folyamatos mozgás számára megállóhelyet, a látványnak kapaszkodót biztosít; egyedül a készítő gondosságán múlik, milyen részletességgel tagolják az egész helyszínt kisebb vagy nagyobb helyiségek sokaságára. A különálló helyeket átjárók kötik össze egymással; megállapodás szerint ezeket égtájakkal szokás azonosítani, így rendszerint tíz-tizenkét irányban haladva kerülhetünk egyikből a másikba (a nyolc égtáj, kiegészítve a FEL/LE/KI/BE lehetőségeivel).

Pl. egy folyópartról észak felé indulva egy erdő pereméhez érkezünk. Egy helyszínről többnyire három-négy kijárat vezet (a többi irányban pl. fal van, vagy más gátló tényezők), s a világ térképét megkaphatjuk úgy, ha ezeket az utakat, mint csomópontokat összekötő hálózatot, lerajzoljuk egy papírra. Hogy ne lehessen rögtön az egészet bejárni, ajtók, kapuk és más nyílászárók feszülnek itt-ott két-két helyiség között (ezek nyitva, csukva és zárva lehetnek), de gyakorlatilag bármilyen típusú akadály is elképzelhető: kőomlás, amit el kell takarítani, ellenséges szereplő, aki elállja az utat, vagy akár egy vizesárok, egy kerítés vagy bármi. A helyiségeket helyhez kötött díszletek és mozdítható tárgyak töltik meg - utóbbiakat föl lehet venni, le lehet rakni, el lehet cipelni máshová és a legkülönbözőbb célokra alkalmazni őket. Tipikus eszközök pl. a lámpa, mellyel sötétben világítani tudunk, vagy a lezárt ajtók kulcsai - de itt is akármi elképzelhető. Ugyancsak jelen vannak itt a hozzánk hasonló (csak a gép által vezérelt) élőlények, szereplők. A legváltozatosabb módokon kommunikálhatunk velük: beszélgethetünk, megtámadhatjuk őket, tárgyakat csereberélhetünk; lehetnek helyhez kötöttek és passzívok, de mozoghatnak is egyik helyiségből a másikba, sőt, hozzánk hasonlóan "saját akaratukból" cselekedhetnek is ezt-azt; lehetnek ellenségesek vagy barátságosak. Legegyszerűbb viselkedésünk egy ilyen világban az lehet, hogy céltalanul csavargunk kusza labirintusának terein, miközben érdeklődve figyeljük előre beprogramozott vagy véletlenszerűen változó életét. Ez minden kalandjáték váza, s különbséget mindössze az tesz egyik vagy másik közt, hogy a helyiségek, akadályok, tárgyak és szereplők egymáshoz való viszonyát mennyire összetetten és sokoldalúan képes megjeleníteni számunkra.

Még néhány szót arról, miért is jelent egy igényes (!!!) szöveges játék elkészítése keményebb megpróbáltatást, mint grafikus testvéreie. Egy igazán intelligens programnak ígék és főnevek százait, ezreit kell megértenie és helyesen alkalmaznia, köztük egy rakás, semmilyen szabályba sem sorolható kivétellel: a játékos ezeket tetszése szerint variálhatja egymással, aminek következtében a parancsoknak sok tízezer vagy akár több millió (vagy még ennél is több!) kombinációja és variációja is elképzelhető bennük, s ez ráadásaként még a helyszínek és helyzetek számával is igencsak fölszoródik (nem mindegy, hogy hol adjuk ki őket, vagy hogy melyik lépésünk előzi meg a másikat) - olyan rengeteg, hogy egyetlen játék(os) sem képes akárcsak töredékéig kiaknázni őket. Ha egy kalandjáték fenn kívánja tartani a valóságosság látszatát (és

ezen keresztül a játékosok érdeklődését...), úgy ezek a lehetőségek közül minél többre kell valamilyen értelmes válasszal szolgálnia, azt az illúziót keltve, mintha birtokában lenne valamennyinek - a programozónak lehetőség szerint minden számbajöhető kombinációt figyelembe véve, elvárnia minden elképzelhető szálát. Később majd látni fogjuk, hogy az utasítások kielemezése, a szórakozás és a szövegértés nehézségei is micsoda egy bonyolult problémával keserítik életünket. Ezzel szemben egy grafikus játék irányítása ikonokon, menükön keresztül zajlik, s a mindössze maroknyi alapfunkció már önmagában töredékére zsugorítja a kipróbálható lehetőségek számát, melyeket csupán listászerűen végig kell venni. (A tényleges megjelenítés problémája pedig már nem is a programozóra, hanem a rajzolóra tartozik...) A cselekvési térnek éppen ez a beszűkülése magyarázza egyébként azt is, hogy igazán intelligens játékokat miért nem lehet grafikusan, hanem kizárólag szöveges alapúként elkészíteni. (Esetleg olyan kereszttevédek formájában, mint amilyen pl. a Maniac Mansion volt, a Times of Lore vagy a Tir Na Nog.) A szöveges játékok legnagyobb varázsát pontosan az nyújtja, hogy sohasem érezhetjük bennük úgy, hogy kimerítettük összes rendelkezésre álló lehetőségeiket; elvégre örökké tömegével maradnak még olyan párosítások, amelyeket egyszerűen nem jutott eszünkbe leírni (különösen, ha a programozók gondoskodtak róla, hogy ébrentartsák bennünk a reményt, mindenféle utalások képében) - egy grafikus játéknál ez a tágasság érzése csak sokkal szerényebb mértékben van jelen. (Azonkívül manapság már egy szöveges játékkal szemben is alapvető elvárás, hogy grafikus képernyőt kezeljen - legalább a megtervezett betűkészlet és az állóképek erejéig.)

3.1.2. Kalandjáték és mitológia

Ha valaki szereti a sört, annak nyilván nagyobb örömet okoz elfogyasztani egy korsónyi Soproni Ászokot, mint a számítógépbe pötyögni, hogy IGYÁL SÖRT. Természetesen a számítógépes játékok nem arra valók, hogy helyettesítsék az igazi dolgokat (helytelenül teszi, aki erre használja őket), hanem hogy valami olyasmit nyújtsanak, amelyet máshol nem talál meg az ember. Mert mi is tulajdonképpen egy kalandjáték?

Világmodell: olyan, mint a Karácsonyfa a gyertyáival, amely a csillagos égre emlékeztet (a boák a Tejutat, az üveggömbök a bolygókat jelképezik rajta stb.). Az általunk elképzelt világnak egy - ugyancsak általunk - létrehozott, kicsinyített mása.

És minél gazdagabb, árnyaltabb és részletesebb ez a világ, annál nagyobb örömet jelent foglalkozni vele. Minden egyes embernek - hacsak nem kíván maga is elveszni, eltévedni vagy szétforgácsolódni a nagyvilágban - alapvetően sürgető igénye van rá, hogy az őt körülvevő világot egy egységes egészként lássa át. Mivel pedig ezt a valóságban kivitelezni - tekintettel rá, hogy a Mindenség milyen elképzelhetetlenül roppant tömegű, szövevényes és bőséges valami nyilvánvalóan reménytelen és lehetetlen vállalkozás volna, így nem marad más hátra, mint ennek mintájára egyszerűsített, stilizált fantáziaképeket, szimbólumokat teremteni magunkban. Olyanokat, melyek - végesek lévén - befogadhatóan és megemészthetően utalnak vissza a Végtelenségre. (Maga a Nyelv, amit beszélünk, a leggyönyörűbb példája ennek!)

A mitológia a régiek által elképzelt hétrétegű világegyetem ábrázolása volt: a regék és mítoszok többsége az egyes csillagképeket és az égitestek mozgását (és azok ránk gyakorolt hatását, a napszakok és évszakok változásait) írja le jelképes, metaforikus megfogalmazásban. A magyar népmeséket elemző Jankovics Marcell írásaiból kiderül, hogy részben ugyanez a hagyomány folytatódik a népmesék világában is: amikor Borsszem Jankó megüli a hatökrös szekeret, az eredetileg a Göncöl hét csillagának megjelenítése volt. (A hetedik, a leghalványabb, a szekér rúdjának a tövében, a Jankó.) Az általunk oly előszeretettel magunkénak vallott Csodaszarvas mondája sem kizárólag a miénk, hanem főként Szibériában, de gyakorlatilag az egész északi félgömbön általánosan elterjedt motívumot alkot; s hogy az éghez kötődik, azt mi sem bizonyítja jobban, mint hogy e terület legtávolabbi vidékein is - egymás létéről semmit sem tudva - megszólalásig hasonló történeteket szöttek az emberek az északi égbolt bizonyos csillagképei köré. (A Cassiopeia, a Perseus és az Auriga együtt egy szaladó szarvashoz hasonlít, s az ezt körülvevő alakok a történet további hősei. A különböző népek sajátos változatai közti különbségek pedig jórészt visszavezethetők arra, ahogy az eltérő földrajzi körülmények miatt másképpen látszanak a csillagok az égen.)

Természetesen nem pusztán szórakozásból fektettek tekintélyes mennyiségű szellemi munkát egy teljes, összefüggő jelképrendszer kialakításába, hanem mert elemi szükségük volt rá: írás-olvasás, naptár, iránytű, térkép és más, tájékozódást segítő találmányok híján ezekkel írták le a körülöttük levő világ naponta, havonta és évenként ciklikusan visszatérő változásait. (Egyes szibériai sámándobokon például olyan elrendezésben lyukakat fúrtak, hogy azokon keresztül az égbolt megfelelő csillagaira nézve, a dob valamelyik díszítése mutatta az északi irányt - úgy használták ezeket, mint tengerész a szextánst.) Nemcsak romantikus álmodozás hozta őket létre, de a maguk környezetében komoly gyakorlati jelentőséggel bírtak. A bűvös számok, alkímiai, asztrológiai jelképek, a népi babonában csökönyösen meglapuló számmisztika -

mindezek talán egy történelem előtti, összefüggő csillagászat és matematika darabokra hullott, törmelékes maradványai.

A szöveges kalandjátékoknak kezdetben még megvolt az az egyedülálló varázsuk, hogy az archaikus világnak erre az elveszített rendjére emlékezhetett általuk az ember (bár persze nem közvetlenül, hanem csak áttételesen: magát az előbbi alapelveit, a csillagképek ábrázolását hiába keresné bennük bárki - de például a kelta mitológia elemeit ezerszer is fölhasználták bennük); a hömpölygő szöveg és a mellette illusztrációként alkalmazott képek látványa régi könyvek hangulatára emlékeztetett; a szellemes útvesztők, a jól eltalált leírások - és egyáltalán: a Nyelv, a Beszéd központi szerepe - pedig tagadhatatlanul némi irodalmi ízt adtak neki.

Azóta lassanként elurulta őket a rajzfilm meg az üzlet, s kiadóik ma már Walt Disney-szerű giccsfigurákkal keresnek dollármilliókat. Amikor még jóformán csak szövegek voltak bennük, a játékosnak (a szerzőnek meg aztán pláne!)

igencsak meg kellett tornáztatnia a képzeletét, hogy maga előtt lássa a mögöttük rejlő világot; és ezt a világot mindenki egy picit másmilyennek látta, attól függően, hogy mennyire és mivel egészítette ki azt önmagában, belül (amitől aztán részben a magáénak is érezhette őket). Ha egy 16-millió színű, térhatású mozgóképpel jelenik meg a képernyőn, az - a tökéletes illúzió lehengerlő erejével bírva - jóval csekélyebb belső részvételt igényel a nézőtől, aki ennek megfelelően nem is csinál egyebet, mint lustán és közönyösen kattintgat az egérrel ide-oda. Ha egy hangzás tökéletes, már nincs mit kiegészíteni rajta, s a hallgatóságra szinte nincs is szükség hozzá: az első pár perc izgalma után méla undorba és unalomba fullad az egész. Mivel a kalandjáték, mint olyan, is csak egy jelképes ábrázolás, ezért egyértelműen a dekadencia jelének tekinthető, ha a játékosok attól vannak elragadtatva, hogy a kép- és hanghatásoknak köszönhetően milyen élethűen jelennek meg benne a főhősök meg a helyszín - ahelyett, hogy a lényegével törődnek a játékosok! A készítőik pedig a divathullámot meglovagolva szorgalmasan csomagolják számukra a látványosan ragyogó semmit...

Jelenleg szinte alig akad olyan területe a PC-s világnak, ahol az eredeti szándék még érvényesülni képes; de ez is inkább csak a technikai körülményeknek (pontosabban azok korlátainak...) köszönheti létét; az Interneten keresztül játszható kalandjátékokról van szó. A TELNET egy viszonylag gyors és egyszerű szöveges párbeszédű kapcsolatot létesít egy központi gép és a hozzá bejelentkező terminálok között - ezt kihasználva rengeteg szöveges játékot írtak és írnak, amelyek egy (rendszerint Linux-os alapú) szolgáltató gépen éjjel-nappal futnak, és a világ bármelyik pontjáról bárki beléjük léphet. Legkellemesebb vonása ezeknek a programoknak az, hogy rajtuk kívül is még akárhány illető részt vehet bennük egyidejűleg, a játék többi szereplőjét is igazi hús-vér emberek irányítják: nincs szükség többé a számítógép által jól-rosszul szimulált értelemre, hiszen valódi értelem (mégpedig egyszerre akár több száz!) jelenik meg helyette, a szereplőkön,

mint szócsöveken át. A különböző játékosok együttműködhetnek, segíthetik egymást, csapatokat alkothatnak, vagy akár versenghetnek is egymással vagy egymás ellen. Ezekben az élet megszakítás nélkül zajlik - hogyha mi kilépünk

belőle, az addigi helyzet akkor is létezik tovább. Található köztük néhány színvonalas alkotás (pl. Holy Mission, Isengard). Hátrányuk viszont, hogy ezeknek a parancsértelmezőjét - lustaságból, vagy ki tudja, miért - szinte minden esetben rendkívül elhanyagolják: sokszor csak a legprimitívebb, "egy ige meg egy főnév" típusú utasítások kiértékelésére képesek, nem lehet egyben több parancsot beírni és így tovább. A hiányzó minőséget itt a mennyiséggel pótolják: a nagyszámú szereplőre méretezve rendszerint sokezer helyszínt és tárgyat zsúfolnak össze beléjük; jóllehet ötleteik többsége nem igazán eredeti. Szintén eléggé rossz tulajdonságuk még az is, hogy sablonos módon valamennyi inkább szerepjáték, mint kaland - ami azt jelenti, hogy különféle ellenségek leöldöcsése árán begyűjtött tapasztalati pontok segítségével lehet fölfelé lépkedni a ranglétrán, és ez a "fejlődés" képezi a játék tulajdonképpeni lényegét és célját. Léteznek belőlük magyar nyelvű alkotások is - egyszerűen borzalmasak. Nem azért, mert a TELNET-en nem lehet keresztülzavarni az ékezetes betűket, ez még rendben is volna; hanem a szövegek hemzsegnek a fogalmazási és helyesírási hibáktól - a parancsok szintaktikájáról nem is szólva... (Egyik legszerencsétlenebb húzás volt például az angol HELP helyettesítése a SÚGÓ-val és hasonlók...)

3.1.3. A számítógép beszélni tanul

A kalandjáték "lelke" - azaz legfontosabb, legbonyolultabb és legtöbbet foglalkoztatott része - a beadott mondatokat értékelő programrész (parancsértelmező, szövegértelmező, értelmező, fordító, vagy angolul "parser", "interpreter" stb.). Hiába a legképrázatosabb szobor-kompozíció, ha az egy bevakolt szobában, a látogatók elől elzárva hever, úgyhogy kívülről senki sem férhet hozzá. Számunkra is egyik leglényegesebb kérdés, hogy a programunkban tárolandó világba miképpen kapjon betekintést a játékos. Egy kritikus

megjegyzés szerint a kalandjátékok a "néma gyerekek anyja sem érti a szavát" közmondáson alapulnak, és ez eleinte valóban így is szokott lenni: ha valaki csak úgy leül egy ilyen játék elé, az leginkább egy üres beviteli mezővel találja magát szembe, és fogalma sincs róla, mit várnak el tőle. A más helyeken általánosságban kialakult szokások ugyan segítenek kissé eligazodni egy új programban is, de legjobb egy teljesen laikus felhasználót feltételezni. Sohasem tudhatjuk előre, hogy a - mienkétől esetleg teljesen elütő észjárású - kalandozónak miféle ötletei támadnak, ezért a játékok intelligenciájának fokmérője, hogy ugyanannak a szándéknak a megnyilvánulásait milyen sokféle különböző formában képesek azonosítani. Például, ha egy ház kijárata egy délkeletre nyíló ajtó, akkor nem elég a programot csak a DK, DÉLKELET, MENJ DÉLKELETRE... stb. típusú parancsok fogadására fölkészíteni, elvégre a szemfüles játékos teljes lelki békével

nyilatkozhat úgy is, hogy MENJ KI A KERTBE, HAGYD EL A HÁZAT, TÁVOZZ, LÉPJ ÁT AZ AJTÓN... - és még hosszan sorolhatnánk ugyanannak a mozgásnak a különféle megfogalmazásait. Minél több ilyen azonos értékű kifejezéssel elboldogul, annál színvonalasabb lesz a programunk, és a játékosok is annál jobban fogják élvezni! Régebben, amikor még a 16-, 48-, 64-kbyte-os mikroszámítógépek jelentették - a játékipiacon - az általánosan elfogadott szintet, elegendő volt azok teljesítőképességére hivatkozni, ami a játékok lehetőségeit is eleve behatárolta. Ma viszont, amikor egy PC nemritkán akár 32-64 MB RAM-ot is rejteget, s a processzorok sebessége is hihetetlen értékeket produkál,

ráadásul mindez folytonosan egyre tovább bővül (és ki tudja, hová vezet mindez?!...), a fizikai korlátok egyszerűen megszűnnek létezni: egy kalandjáték színvonalát most már egyedül a készítője lustasága, vagy fantáziája, szókincsének bősége (vagy prózaibb okok közül: a készítésre rászánt idő...) határozza meg. De tény, hogyha egy ÜSD LE A RABLÓT mellett azt is magunkénak akarjuk tudni, hogy LÁSD EL A BAJÁT A RABLÓNAK, az nemcsak egyszerűen a szótár bővítését jelenti, hanem efféle szóvirágok kezelésével a kód is egyre bonyolultabbá válik...

Na de túlságosan előre szaladtunk (ez lett volna a bevezető ugyanis...): hogyan is szólunk egyáltalán egy ilyen játékhoz? Valamit ugyebár cselekedni óhajtunk benne, és ezt a közvetítő programnak az értésére próbáljuk adni: szándékunkat egy megállapodás szerinti nyelv szabályain keresztül kell megfogalmaznunk. Legjobb (volna), ha ez a nyelv mind közelebb áll (na) a természetes beszélt nyelvhez - de ahhoz, hogy ezt maradéktalanul megvalósítsuk, egy olyan "számítógépet" kellene építenünk, melyben - mint régi szélhámos sakkautomatákban - benne ül egy ember... Megközelíteni azonban lehet - pontosabban azt a látszatot kelteni, mintha megközelítettük volna.

Általános szabály, hogy mindinkább egyszerűnek, spontánnak és magától értetődőnek látszik valami a felhasználó szemszögéből nézve, annál nagyobb apparátust kénytelen a programozó megmozgatni hozzá, hogy egyáltalán működjön

a dolog. A kívülálló számára természetes a nyelv, amelyen beszél, a legcsekélyebb szellemi erőfeszítés nélkül helyezi egymás mellé a szavakat.

Fogalma sincs róla, milyen egy irtatlanul bonyolult rendszernek a birtokában van, s hogy a szerencsétlen programozó mennyit izzad vele, míg végül valami, ehhez úgy-ahogy hasonlót nagynehezen kiprésel magából. Különösképp érvényes ez a magyar nyelvre. Világszerte a kalandjátékok túlnyomó többségét angol nyelven írták. Ez nemcsak a nyugati régió egyik legáltalánosabb nyelve, de egyszersmind a legegyszerűbb nyelvtani szerkezetűek közé is tartozik - ezért is olyan könnyűszerrel megtanulható, mint köztudott róla. Az angol nyelv nem (vagy csak elvétve) használ ragokat, jeleket, maguk a szavak többnyire változatlanok maradnak, helyette a köztük fennálló viszonyt az erősen kötött szórend és az elöljárók mutatják meg. A főneveknek nincsenek nemeik, és még csak a magázódást sem ismeri. Összetett szavak képzése is úgy történik, hogy egyszerűen egymás mellé raknak két darab névszót, s legtöbbször még csak egybe sem írják őket (vagy ha mégis, akkor kötőjellel). Semminemű ékezetet nem használnak, az összetett hangzókkal sincs különösebb gond. Egyszóval, akár csak az akácfa vagy a káposztalepke, az angol nyelv is alapvető igénytelenségének köszönheti azt, hogy világszerte elterjedhetett... Nézzük meg, hogyan hangzik angolul, ha valakit megkérünk rá, hogy nyissa ki az ajtót: OPEN THE DOOR (esetleg még hozzátehetjük, hogy PLEASE - de ennek most semmi jelentősége nincsen). Az egész felszólítás két egyszerű szóból tevődik össze (a névelővel most nem foglalkozunk, azokat szimplán át lehet lépni): OPEN és DOOR, melyek - bármilyen összefüggésben is használjuk őket - minden esetben változatlanok maradnak, nem számítva az időnként a végükre kerülő "-s" betűt vagy "-ing" végződést, így egy szótárból kikeresve szemvillanás alatt azonosítani lehet őket. Ráadásul a mondat szórendje is szigorúan kötött (nem mondhatjuk pl. azt, hogy DOOR OPEN), tehát bizonyosan tudjuk róluk azt is, hogy az első az ige és a második a főnév. (Itt most csak a kalandjátékok irányításához felhasznált nyelvről esik szó.) Bővítsük most egy határozóval ezt a mondatot! Tegyük fel, hogy kulccsal szeretnénk kinyitni az ajtót. Ez esetben így hangzik: OPEN THE DOOR WITH KEY.

A szórend most is kötött, és mindössze annyi a különbség, hogy a WITH elöljárót követő főnevet, mint eszközhatározót vesszük figyelembe, nem pedig, mint tárgyat.

Ugyanez magyarul már jóval keményebb diót jelent. Hogyan is fordítanánk le az előbbi példákat úgy, hogy értelmesen összefüggő mondatokat kapjunk? Valahogy így: NYISD KI AZ AJTÓT (ha netántán magázódunk a programmal, akkor a NYISD helyett NYISSA áll...), vagy ha nem parancsolgatni akarunk, akkor mondjuk KINYITOM AZ AJTÓT. De mi van akkor, ha NYISD KI AZ AJTÓT KULCCSAL helyett mi inkább úgy fogalmaznánk, hogy NYISD KI KULCCSAL AZ AJTÓT?

Vagy: KULCCSAL NYISD AZ AJTÓT KI? Esetleg: AZ AJTÓT A KULCCSAL NYITOM KI?... Folytathatnánk a sort: valamennyi értelmes, és - a hangulati árnyalatoktól eltekintve - ugyanazt jelentik. A szórend itt már semmiféle támpontot nem

nyújt, hiszen gyakorlatilag korlátlanul szabad - a szavak értelmét egymástól elszakítva, önmagukban kell megtalálnunk. De miképpen döntsük el, hogy egyáltalán melyik szóról van szó? Hiába szerepel a szótárunkban a NYIT ige, ha egyszer nekünk olyanokat írnak, hogy NYISD, NYISSA, KINYITOM, NYITOM... - és még vagy százféle különböző ragozott alak. Még ha attól eltekintünk is, hogy az igekötővel egybe- avagy különírjuk-e az igét, akkor is mennyi lehetőség marad! A főnevek ragozása ugyancsak egy cifra eset. Már maga a legalapvetőbb tárgyrag sem pusztán csak egy "-t" betűből áll: magánhangzós végű szavaknál ékezetet kap az utolsó betű (pl. alma - almát), néha kimarad egy hangzó (bokor- bokrot, tükör - tükröt), hosszú magánhangzók rövidülhetnek (ég - eget, tűz - tüzet), azután meg ott vannak a különféle kötőhangok is (-at, -et, -ot, -öt), minden szóhoz más-más fajta. Minden egyes határozónak saját ragja van: -ba, -be, -ra, -re, -n, -on, -en, -ön, -ból, -ből, -ról, -ről, -tól, -től, -nak, -nek, -hoz, -hez, -höz, -val, -vel, -vá, -vé, -kor, -ért, -ig... stb. stb.

Ebből is van vagy negyven-ötven féle! (A KULCCSAL ráadásul még egy különleges eset is, hiszen -val, -vel esetén a "-v-" helyett duplázódik az utolsó betű, s miután "cs"-ről van szó, még csak nem is az "s", hanem a "c" betű...)

Amennyiben emellé még a többesszám és birtokos eset jeleit is engedélyezzük, úgy az előbbi mennyiség ezek számával szorozódik (!!!), tehát végeredményben minden egyes főnévnek több száz féle különböző ragozott alakja lehet! Ha a

szótárunk mondjuk ezer szóból áll, akkor a kereséskor ez több százezer szóval való összehasonlítást jelent, ami már önmagában véve is egy képtelenség; nem beszélve arról, hogy lehetetlen volna ennyi alakot mind-mind letárolni.

Megabyte-okat foglalna el, és percekig tartana megtalálni benne valamit...A legegyszerűbb, korai, igénytelen játékokban még kikerülték ezt a problémát, ahelyett, hogy megbirkóztak volna vele. Valamiféle hallgatóságos megállapodással kialakítottak maguknak egy ún. "csonka magyart" - ezalatt a nyelvnek egy roppant alacsony szintre csökkentett változata értendő. Az angol forráshoz igazodva, azonos sorrendű szópárokat várt el bevétel gyanánt a program, és kizárólag a szavak ragozatlan szótári alakját lehetett alkalmazni benne. Az idézett parancsok így hangzottak ezen a nyelven: NYIT AJTÓ, ill. NYIT AJTÓ KULCS... Némelyest fífikásabb, de éppilyen gépies és felszínes megoldásnak bizonyult az is, amikor a szavak első néhány betűje alapján kerestek a szótárban (többnyire az első háromtól az első ötig terjedt ez a hossz). Például egy program az első négy betűt vette figyelembe, s ezáltal a NYISS ABLAKOT utasításból ennyit látott csak: NYIS ABLA. A szótárban ennek alapján egységesen négybetűs jelsorozatok voltak tárolva, az ennél rövidebb szavaknak - vagy amelyek töve megváltozott, mint a NYIT ige esetében a "t" és az "s" betű - természetesen föl kellett venniük a ragozott alakjait is. (A CSOMÓ és a CSOMAG szavakat már nem is tudta volna megkülönböztetni egymástól.)

Amennyiben igényesebb programokat szeretnénk írni, nem elégedhetünk meg ezekkel a primitív mankókkal! Meg kell tanítanunk beszélni azt a buta ócskavast...

Szükség lesz tehát egy algoritmusra, mely a szótárban külön-külön meglévő szótövek és ragok alapján dolgozik, és mindkét irányú működésre képes: egyrészt a begépelte szavakat szétbontja szótövekre és ragokra (ezeket a programon belül majd sorszámmal fogjuk azonosítani), vagyis értelmezi azokat - másrészt fordítva: a megadott szónak előállítja a megadott típusú ragozott alakját. (Erre azért lesz szükség, hogy parancsunkra a számítógép válaszolni is tudjon, azaz ki tudja írni helyesen a képernyőre a neveket.) Figyelembe kell vennie bizonyos szavak különleges tulajdonságait is (épp az imént említettük őket: kulccsal, tükröt, nyisd...), és különbséget kell tennie az igék és főnevek teljesen eltérő ragozása közt. Fontos követelmény még, hogy megfelelően gyors legyen: elvégre a legegyszerűbb mondatok is min. három-négy szóból állanak, de mint később látni fogjuk, ennek akár a sokszorosát is kaphatjuk - a szótár pedig nyugodtan állhat akár több ezer szóból is.

3.1.4. A szavaktól a mondatok felé

A kalandjátékok utasításaira általában jellemző, hogy az alany hiányzik belőlük ill. rejtett (első vagy második személyű), s csaknem kivétel nélkül valamilyen igei állítmányra épülnek, amihez gyakran különféle tárgyak, jelzők és határozók csatlakozhatnak. Utóbbiak önmagukban egyszerűek, egy-két szóból állanak, s ami még nagyon jelentős egyszerűsítés, hogy alá-, fölérendelt tagmondatok sosem fordulhatnak elő. (Legalábbis ezidáig nem volt rá még példa...) Tipikusan egyszerű parancs pl. az alábbi: VEDD FEL A KULCSOT.

Ebben föllelhető mindkét alapvető szófaj: az ige és a főnév - tulajdonképpen az összes többi szófaj fölfogható úgy is, mintha ezek speciális esetei lennének. Az ige fontos tartozéka (vagy inkább: része) még az igekötő is, hiszen ha azt mondjuk, hogy VEDD FEL, VEDD LE vagy VEDD MEG, azok teljesen eltérő cselekvésre utalnak. E kettő együttesen dönti el, hogy mit akarunk csinálni, a főnevek és társaik pedig a cselekvésünk célját határozzák meg, azáltal, hogy az elérhető tárgyak és szereplők közül konkrétan kijelölnek valamit. Ez nem mindig egyértelmű: egyrészt létezhet a játékban több hasonló nevű tárgy is, másrészt használhatunk különféle gyűjtőneveket. Előbbi esetre példa lehet az, ha többféle különböző kulcs van: ilyenkor mindegyiknek van valamilyen sajátos jelzője, mely megkülönbözteti az összes többitől (pl. "kis kulcs" vagy "nagy kulcs" stb.); a főnevek elé melléknevek járulnak (VEDD FEL A NAGY KULCSOT). Gyűjtőnév esetén pedig az illető szó ugyan pontosan megjelöli a tárgyat, ám egyszerre akár többet is - legjobb példa erre a MINDEN szónak az alkalmazása: egy VEGYÉL FEL MINDENT utasítás egyszerre vonatkozik a közelünkben látható valamennyi tárgyra. Melléknevek szintjén is elképzelhető hasonló, pl. így: VEDD FEL AZ ÖSSZES KULCSOT. (Hogy miért is célszerű a melléknevet is speciális főnévként nyilvántartani, az kitűnik abból, hogy alkalomadtán ezek is a főnevekhez hasonlóan ragozódhatnak. Egy jólnevelt kalandjáték ugyanis, ha nem egészen biztos a dolgában, további kérdéseket tesz fel: megkérdezi tőlünk, hogy melyik kulcsot szándékozunk fölvenni. Erre mi felelhetjük azt is, hogy a NAGY KULCSOT - vagy egyszerűen ennyit: NAGYOT. A melléknév ilyenkor kifejezetten úgy viselkedik, mintha főnév lenne; fordított esetben pedig egyes főnevek is állhatnak a melléknévi jelzők helyén.) Mivel igencsak ritka eset (úgy értem, egy kalandjátékban...) az, hogy teljesen azonos tárgyakból legyen több példány ugyanazon a helyen, s ráadásul ezek közül többre, de nem valamennyire akarunk hivatkozni egyidejűleg, ezért számnevek csak elvétve fordulnak elő; ilyenkor, mint sejthető, a melléknévhez hasonlóak, csak funkciójuk más. Ugyancsak fehér holló egy másikfajta jelzős szerkezet, nevezetesen a birtokos jelző: egyfelől erre is csak viszonylag ritkán kerül szükség, másfelől meg akkor is kiválthatjuk más, egyszerűbb jelzőkkel a szerepét. Pl. ha az iménti nagy kulcs egy űr kezében van: egy VEDD EL AZ ŐR KULCSÁT helyett a FOGD A NAGY KULCSOT utasítás sokkal egyszerűbb. Jóllehet ezek kezelése eléggé bonyolult eljárást igényel, mindenesetre sokat emel egy játék színvonalán, ha - mint értékes egzotikumot - ilyesmit is beleépítünk. Annál nagyobb igény lehet ezzel szemben (márminthogy a játékosok oldaláról) egy másik, teljes főnévi értékű szófajra, a névmásra: ha leírjuk, hogy BESZÉLGESS VELE vagy NÉZZ RÁ, miközben a legutóbbi parancsunk egy KÖSZÖNŐ A MADÁRIJESZTŐNEK volt, ez kétségkívül azt jelenti, hogy ezúttal is a madárijesztőre gondolunk - az utolsóként előfordult főnevet helyettesíthetjük valamilyen névmással. Érthetetlen, hogy ezt is csak alig néhány program ismeri, amikor pedig a megvalósítása egyszerű (hisz' mindössze egy szócskeréről, behelyettesítésről van szó), és igen nagy könnyebbség, ha nem kell a hosszadalmas MADÁRIJESZTŐ szót mindig újra és újra bepötyögni, hanem helyette ezt a "kézreálló" pár betűt...

A határozók már nem önálló szófajok, hanem csak mondattani egységek - lényegében a főnevek eltérő ragozású alakjai. Miután a szórend szabad, és más támpontunk nincsen, egyedül a ragozásuk alapján tudjuk csoportosítani a különféle kifejezéseket. Túlságosan körülményes és fölösleges is volna azonban valamennyi raghoz külön határozót rendelni, így legcélravezetőbb öt csoportba sorolni őket (az egy csoportba tartozó ragok nemigen szoktak előfordulni egy parancsban együtt, és körülbelül hasonló dolgot jelentenek, így nyugodtan tekinthetjük azonosnak őket): 1.) tárgy: -t raggal vagy rag nélkül; 2.)

helyhatározó: -n, -ban, -nál, -ról, -ból, -tól; 3.) célhatározó: -ra, -ba, -nak, -hoz; 3.) eszközhatározó: -val; 5.) egyéb: -ért, -ig... stb. Pl. az ADJ PÉNZT A KOLDUSNAK utasítás három mondatemet tartalmaz: egy igét, egy tárgyat és egy célhatározót (az, hogy ezt a nyelvtanban részeshatározónak nevezik, ne zavarjon túlzottan bennünket...). A ragokon túl használhatunk névutókat is a főnév jelentésének módosítására, így helyhatározó lehet az is, hogy ASZTAL ALATT, vagy a KULCCSAL helyett eszközhatározó, hogy KULCS RÉVÉN vagy KULCS SEGÍTSÉGEVEL stb. stb.

Összefoglalva tehát, minden egyes begépelte parancs lényegében a következő részekből tevődik össze: az ige (a beleértett igekötőjével együtt), valamint a felsorolt ötféle tárgy ill. határozó. A parancsot akár fölfoghatjuk úgy is, mint egy rekordot, aminek ez a hatféle mezője van. A programon belül persze a szavak nem szöveggé jelennek meg, hanem minden egyes dolgot valamilyen sorszám azonosít. Pl. ha a "kinyitni" ige a 13-as sorszámot kapta az előzetes tervezés során, akkor ez azt jelenti, hogy az ige változójába egy 13-as szám kerül. Amelyik fajta eleme hiányzik a mondatnak, az természetesen a 0-ás értéket kapja. Ez gyanúsán egyszerűnek látszik, holott valójában egyáltalán nem az: ugyanis ez a végeredményül kapott egyszerűség a valóságban egy hosszú és bonyolult értelmezési folyamatot takar, melynek folyamán a különféle szókapcsolatok és más szerkezetek fokozatosan a lehető legtömörebb viszonyokká bomlanak le. Ahhoz, hogy ez megvalósuljon, tucatnyi eltérő algoritmust kell egymás után lefuttatni, amelyek mindegyikének feladata a megfelelő szavak vagy szópárok cseréje, helyettesítése valamilyen másik, egyszerűbb szóval. Ezek a cserék táblázatokban való keresések útján történnek. Efféle algoritmus lehet például az, amelyik az igék és igekötők különálló párosát egyetlen összetett igére cseréli ki (a NYISD és a KI szavakat törli, és helyükre egy KINYITNI szót ír - ha pedig nemlétező párosítást talál, pl. azt írjuk, hogy NYISD ÖSSZE, ami nincs benne a táblázatban, akkor hibát jelez, és az ige értelmezhetetlen). Vagy egy másik, mely a névmás helyére elegánsan becsempészi az utolsóként alkalmazott főnevet. Egy olyan, mely a "melléknév + főnév" szókapcsolatot egyetlen új főnévvel helyettesíti be (a NAGY és a KULCS szavak sorszámait kiiktatva, helyettük egy másik, a NAGY KULCS szó sorszámát illeszti a mondatba). És így tovább; minél több ilyen funkcióval ellátjuk, annál intelligensebb lesz a programunk. (És annál nehezebb lesz megírni...) Természetesen az sem mindegy, milyen sorrendben hajtjuk végre ezeket az egyes szabályokat a mondatban. Pl. ha azt találjuk leírni, hogy NAGYOT, akkor az először is felbomlik a - kissé idegenszerűen hangzó - NAGY AZT szókapcsolatra, majd az AZ helyére bekerül a KULCS főnév (mint névmás-helyettesítés!), s csupán ezután lehet belőle NAGY KULCSOT. Egy szöveges kalandjáték értelmező programját elkészíteni körülbelül hasonló feladat, mintha mondjuk egy PROLOG fordítóprogramot óhajtanánk írni...

A parancs és a mondat kifejezéseket az eddigiekben egy kissé összekeverten használtuk, úgyhogy lassanként ideje lenne szétválasztanunk őket

egymástól: míg egy parancs mindig egyetlen meghatározott cselekvésre utal, vagyis lényegében tagmondat (a fentiekben hozott példák mind parancsok voltak), addig egy mondat nyugodtan akár több parancsból is állhat. (Már amelyik program hajlandó ezt tudomásul venni...) Vagy az ellenkező irányból megközelítve a dolgot: egy mondat az a szövegmennyiség, amit a játékos egy szuszra begépel a bevétel során, s a parancs ennek egy-egy önállóan végrehajtható része. A mondaton belüli parancsok elválasztására írásjeleket (pont, vessző és társaik), valamint az ÉS szócskát vagy annak szinonimáit (MAJD, TOVÁBBÁ, VALAMINT... stb.) használjuk. A játék a mondatban lépésről lépésre halad előre; mihelyt egy parancsot végrehajtott, utána kezdi el értelmezni a következőt. Pl. egy tipikus összetett mondat így hangzik: VEDD FEL A KULCSOT ÉS NYISD KI VELE AZ AJTÓT. Ebben az esetben a két tagmondatot akár egy-egy külön mondatként is le lehetne írni, de előfordul olyan is, hogy szorosabb összefüggés kapcsolja őket össze, miáltal szétválaszthatatlanok lesznek - ez akkor állhat fenn, mikor a második parancsból kihagyunk valamit,

amit már tartalmaz az első. Pl.: GYÚJTSD MEG A GYERTYÁT ÉS A MÉCSEST. Itt a második parancsból kimaradt az ige. Ahhoz, hogy ezt értelmezni tudjuk, egy aprócska trükköt kell alkalmazni a programban, nevezetesen amíg a mondatnak nincs vége, addig az aktuális parancs változóit sohasem töröljük, hanem csak az újabbik parancs szavai mindig felülírják a régit - így tehát a teljes első parancs változatlanul megmarad, mindössze a MÉCSES, mint tárgy fogja felülírni az előző tárgyat, a GYERTYÁ-t. Ezáltal egyetlen állítmány mögé tetszőleges számú tárgyat felsorakoztathatunk. Sokkalta nehezebb volna - éppen hogy a lépésenkénti végrehajtás miatt! - ugyanezt az ellenkező irányból is megcsinálni, valahogy ilyképpen: NYISD KI ÉS CSUKD BE AZ AJTÓT. Ebben az esetben ugyanis a programnak szinte "előre kéne gondolkodnia": amikor odáig jut, hogy NYISD KI, még fogalma sincsen róla, hogy mire is vonatkozik ez a dolog - jóllehet a keresett főnév benne van a mondatban, csak jóval később volna esélye megtalálni azt. Az eredmény: a számítógép megkérdi, hogy "Mit akarsz kinyitni?". Ennek feloldására be lehetne vezetni egy ún. öröklési rendszert, ami abból állna, hogy még mielőtt hozzálátna a legelső lépés végrehajtásához, a program végigértelmezné a teljes mondatot, és mindegyik tagmondat állapotát eltárolná valahol, egy rekordokból álló láncban úgy, hogy az aktuális rekord mindig örökölné - az iméntiekben leírt módon - az előzőnek azt a részét, ami nem változott. Azután ugyanezt végigcsinálná fordított irányban, az utolsótól az első felé haladva is - és ami valahonnan hiányzik, azt mindig a másiktól vett megfelelő részekkel pótolná és kiegészítené benne ekkor is. Eközben az első parancsba, ahonnan hiányzik a tárgy, beíródna az utána következő tagmondat tárgya (ami esetleg szintén a következőből örökölte azt, és így tovább). Csupán ezt követően hajtódna végre a parancsok. Amivel aztán el is érteztünk az értelmezés következő állomásához: mi történjék, amikor valamilyen szó végképp hiányzik a mondatból. Erre több megoldás is kínálkozik. Lehet egyszerűen a felhasználó képébe vágni, hogy márpedig ez nem egy értelmes mondat, és legközelebb szíveskedjék jobban megválogatni a szavait... Ennél valamivel udvariasabb (és intelligensebb) megoldás, amikor a homályos részekre vonatkozó kérdéseket teszünk fel, és további kiegészítéseket várunk. Pl. ha csak annyit kaptunk parancsként, hogy NYISD, akkor feltesszük neki a következő kérdést: "Mit akarsz kinyitni?" - ami után következő bevitelként kétféle felelet várható. A játékos vagy begépel egy teljesen új mondatot, nagylelkűen elfelejtve az előzőt - vagy pedig a kérdésünkre válaszol, és mindössze a hiányzó tárgyat adja meg. Hogyan készítsük fel a játékot a helyes reagálásra mindkét esetben, amikor e kettőt tulajdonképpen nem is igazán lehet megkülönböztetni egymástól? Hát úgy, hogy nem is különböztetjük őket meg. Az előbb láttuk, hogy az esetleg összefüggő részmondatok miatt a parancsok változóit mindig a mondat végén töröljük, amikor egy új mondatba kezdünk. Egyszerűen annyit kell tennünk, hogy mikor valamilyen kérdést intézünk a kalandorhoz, olyankor az újabb bevitel megkezdése előtt nem töröljük ezeket a változókat, azaz nyitva hagyjuk a mondatot - akármit begépel a tisztelt felhasználó, az a jelenleg meglévő parancsot fogja felülírni, éppúgy, mintha az a következő parancs volna. Teljesen mindegy, hogy töredékeket kapunk-e tőle, vagy egy új mondatba kezd. Így akár több, eltérő tartalmú kérdés is követheti egymást, és mégis valamennyi válasz ugyanazt a parancsot fogja fokozatosan gazdagítani - szabályos kis párbeszéd alakulhat ki a program és a felhasználója között. Komolyabb probléma akkor adódik, ha már bevezettük az előzőekben vázolt öröklési rendszert - ilyenkor ugyanis a kérdés feltevése előtt meg kell jegyeznünk az egész mondat állapotát, majd a kiegészítés megtétele után ismételtlen végigfuttatni rajta oda-vissza a parancsokat összefűző eljárást. Harmadik, és egyben legintelligensebb megoldása a hiányos mondatok esetének, ha a program maga automatikusan megkísérel behelyettesíteni a kimaradt részek helyére valamit, és ha csak ez végképp nem sikerül neki, akkor teszi fel a kérdést. Pl. a NYISD megadása esetén sorjában végignézi az összes elérhető tárgyat, s csak ha nincs közöttük egyetlen olyan sem, ami nyitható,

akkor kérdi meg a kezelőjét, hogy voltaképpen mire is gondolt. (Vagy ha többet talált, akkor azt, hogy melyiket.) Hasonlít ez némiképpen a gyűjtőnevek kezeléséhez. Leggyakoribb gyűjtőnév a korábban már megemlített MINDEN szó: ilyenkor a parancs végrehajtását egy külső ciklusba kell ágyazni - sorban egymás után behelyettesíteni a kívánt szó helyére az összes elérhető tárgynevet, és mindegyikkel egyesével végrehajtani ugyanazt a parancsot. (Érdeemes elgondolkodni rajta, hogy mi történik akkor, ha netán a huncut játékos egyazon parancson belül többször is alkalmazott valamilyen gyűjtőnevet. Pl. így: MUTASS MEG MINDENT MINDENKINEK. Ez az, ami a programozó idegeit végképp próbára teszi...)

3.1.5. Helyiségek összefüggő labirintusa

Említettük, hogy a kalandjáték életének teljes színtere egyes, különálló helyiségekre tagozódik. Ez valahogy úgy néz ki, hogy tartózkodunk valahol, pl. egy szobában, és érzékelésünk azokra a dolgokra korlátozódik, amelyek szintén ugyanazon a helyen vannak - egészen addig, míg át nem haladunk valamelyik szomszédos területre. Hogyha a jelenlegi szobában hever egy asztal, akkor azzal, mint tárggyal, elméletileg megtehetünk bármit (megvizsgálhatjuk és hasonlók), de ha valahol másutt leírjuk, hogy ASZTAL, akkor már az alábbihoz hasonló válasz érkezik: "Nincs itt semmiféle asztal." Miképpen lehet ezt a világot a számítógépen keresztül leképezni a játékosok számára? Minden helyiség egyértelműen azonosítható és a többitől megkülönböztethető kell legyen, és a játékban szereplő valamennyi objektumot (tárgyat, élőlényt, beleértve minket is!) pontosan el kell helyezni valahol a térképen, úgy, hogy annak helyzete változtatható is legyen. Le kell írni a csatlakozó helyiségek viszonyát, a mozgást gátló akadályokkal és más tényezőkkel együtt. Természetesen hűek maradunk eddigi alapelvünkhöz: mindenhol sorszámok és táblázatok sarjadnak a lépteink nyomán. Egy jól szervezett kalandjáték-világ valójában igen összetett rendszert alkot, annak megfelelően, amit a filozófiában is előszeretettel hangoztatni szoktak: minden összefügg benne minden egyébbel... Emiatt is olyan nehéz eldönteni, hogy egyáltalán hol kezdjen neki az ember. A parancsok és cselekvések végrehajtásának kidolgozásába bele sem foghatunk addig, amíg a játék teljes leendő világának leíró adatai holmi adatbázis-szerűen nincsenek letárolva a gépben. Egy nagyobb lélegzetű műnél ez hosszú hónapokig tartó tervezést és előkészületeket igényel, mielőtt még egyetlen sort is leírnánk bármilyen programozási nyelven. Igazából ez a munka legkellemesebb része, hiszen - játékról lévén szó - jórészt álmodozással, ötleteink lejegyzésével és összerendezésével, az epizódok és jelenetek kiagyalásával telik el. Azonban fabatkát sem érne az egész, ha nem mindjárt abban a formában kezdenénk el az adatok gyűjtését, ahogyan azt majd a későbbi programban felhasználni fogjuk... Legelső feladatunk, hogy fölépítsük a helyiségek rendszerét. Ehhez legjobb, ha fogunk egy darab üres papírost, és térképet rajzolunk a képzeletünkben fölsejlő világról. Ez a térkép egy csöppet elvont megjelenéssel bír a szokásos atlaszokhoz képest: a helyszíneket vázlatos körök jelképezik, az egyikből a másikba vezető utakat a közéjük húzott vonalak. Mindegyik helynek valamilyen nevet adunk (pl.: "Erdei tisztás"), ezt a megfelelő karika belsejébe írjuk; s hogy két-két helyiség között melyik irányban lehet közlekedni, azt a köztük feszülő vonal végére biggyesztett nyállal ábrázolhatjuk. A nyíl tövébe írjuk azt is, hogy mi módon kell mozognunk a jelzett helyváltatáshoz: ennek jelzésére - megállapodás szerint, de persze nem kötelező jelleggel - az iránytű nyolc égtája valamint a két függőleges irány szolgál. Pl. ha a lakásunk szobáit szeretnénk ábrázolni, ami öt helyiségből áll, akkor a papírra öt darab kört húzunk, a szobák állásának megfelelő elrendezésben, beléjük írálva egy-két szavas elnevezéseiket; ha

mondjuk a hálószoza a nappalitól nyugatra fekszik, akkor a "Nappali" feliratú buboréktól a "Hálószoza" feliratúig rajzolunk egy nyilat, tövében egy NY betűvel, s miután nyilván visszafelé is vezet út, a vonal ellenkező végét is nyíllal látjuk el, a másik tövébe egy K betűt írva. Ha a két szoba közt ajtó is nyílik, a vonal közepe táját kiegészítjük egy kisebb téglalappal, belevezetve az "Ajtó" szót... Azután ezt a módszert végigvisszük a termék egész hálóján. (Tág kiterjedésű, szabadtéri helyeknél élhetünk azzal a trükkel, hogy önmagába visszamutató nyilakat rajzolunk - ameddig a játékos a jelzett irányban gyalogol, folyton ugyanabban a térben marad.) Amerre egy helyről nem vezetnek nyilak, ott fal van, vagy más hasonló áthághatatlan akadály. Nagyobb térképeket célszerű több részre bontva elkészíteni, hogy áttekinthetőek legyenek.

Idáig minden pofonegyszerű, bárkinek eszébe juthatott volna mindez - csak kár, hogy a számítógép egy összefirkált papírlappal semmit sem tud kezdeni. Le kell fordítanunk a számok nyelvére az egészet. Kezdjük el ezt azzal, hogy valamennyi helyünknek egytől növekvő rendben sorszámokat osztunk: a hálószoza lesz az 1-es, a nappali a 2-es stb. - írjuk be ezeket a megfelelő buborékokba, a névszövegek mellé. A választható irányokhoz is rendeljünk hasonlóan sorszámot; a felsorolt tíz égtáj mellé fölvehetjük még a KI és BE irányokat is, úgyhogy összesen tizenkét irányunk lesz; 1-től 12-ig számozzuk meg őket - mondjuk az 1-es jelentse az északot, és így tovább. Ezt követően a térképet már táblázattá lehet alakítani: olyan elrendezésben, ahol baloldalt függőlegesen a helyiségek sorszámaikat tüntetjük fel sorban, vízszintesen pedig az irányok helyezkednek el. A táblázatban minden helyhez tartozik egy sor, ezeket úgy kell kitölteni, hogy a sor mindegyik rubrikájába annak a másik helyszínek a száma kerül, ahová a hozzá tartozó irányban az adott szobából kimozdulva érkezünk. Ha a jelzett irány sehová sem vezet (lezárt irány - fal), akkor ide nullát írunk, ha önmagába kanyarodik vissza, akkor a saját helyszín száma kerül ide is. Ha arra vagyunk kíváncsiak, hogy a nappalitól (ez most ugye a 2. helyiség) észak felé (1. irány) mi található, akkor egyszerűen kiolvassuk a táblázat 2. sorának 1. oszlopát, ahol tegyük fel, egy 4-es számot találunk (legyen ez az előszoba száma): a nappalitól északra az előszoba vár ránk. Ezt számítógéppel kezelni igen egyszerű lesz később.

3.1.6. Barangolás a térképen

Mihelyt ezzel megvolnánk, ne felejtünk még egy második táblázatot is készíteni, az egyes utakat lezáró akadályok nyilvántartására. Mondanom sem kell, hogy ezek is tipikus sorszámokat kapnak, ám itt már nem árt némi rendszert is vinni a dologba: ha valamiből több hasonló van, úgy célszerű azokat egy csoportba venni, még ha távol esnek is egymástól (pl. 1-től valameddig számozzuk az ajtókat, utána jönnek sorjában az ablakok stb.); ezáltal a későbbiekben megkönnyítjük ezek kezelését. Az egy-egy akadályhoz tartozó sorokban itt a következő adatokat szükséges feltüntetnünk: két-két rubrika az általuk elválasztott két-két helyiség számára (pl. a nappaliból a hálóba nyíló ajtó esetén egy 1-es és egy 2-es), valamint egy harmadik mező, ami pedig az akadály jelenlegi állapotát tükrözi. Ezúttal is számokkal kell élnünk: legalább három értéket használjunk (célszerűen 0-tól 2-ig számozva) annak jelzésére, hogy az ajtó nyitva, csukva vagy zárva van-e - de ennél jóval több lehetőség is elképzelhető (lehet mondjuk félig behajtván vagy betörve az ajtó, vagy függöny esetében behúzva vagy elhúzva, akármi az eszünkbe juthat!). Végül hagyjunk még egy negyedik oszlopot is saját megjegyzéseink részére: ide lehet beírni az akadály nevét (ajtó, kapu stb.), és esetleg még azt is, hogyan kezelődjenek később, azaz miképpen nyitható (milyen cselekvést kell véghezvinni az állapota megváltoztatásához - mondjuk milyen kulccsal nyitható egy zárt ajtó). Listánkba fölvehetünk olyan "elvont" akadályokat is, mint egy kerítés, vagy az utunkat álló eleven ôrszem - csak vigyázzunk ez utóbbi esetén, mert ô

az ajtókkal ellentétben nem két, hanem egyszerre csak egy helyiségben léteznek, s így másként kell majd kezelni később. Egyáltalán, az akadálylistába fölvenni azokat a tényezőket érdemes, amelyeknek állapota változik - ha egy falon sohasem juthatunk át semmilyen módon, azt egyszerűbb inkább a másik táblában a megfelelő irányhoz, mint valami fiktív, valójában nemlétező helyszínszámot beírni.

Most már megtehetjük első - egyelőre csupán képzeletbeli - sétánkat kezdetleges szobáink kis színpadán. Azt is látjuk, hogy - noha még csak két táblázatunk van - máris mennyi mindent kéne ellenőriznie egy vezérlő programnak olyan, primitívnek látszó művelet elvégzéséhez, mint egy meghatározott irányban való haladás. Először is - jelenlegi pozíciónk tudatában - a bejárás táblázatból ki kell olvasnia az elérendő célhelyiség számát. Meg kell néznie, az akadályok listájában szerepel-e olyan objektum, melynek összekapcsolt két helyisége éppen az általunk igénybe vett két szobával egyezik-e meg (ráadásul duplán is kell vizsgáldnia, mert hátha az ellenkező irányból jövünk, a 2-esből az 1-esbe, s nem pedig az 1-esből a 2-esbe), majd ha talált egy ilyet, akkor tájékozódni róla, nyitva van-e az akadály. Ha nem járható az az út, figyelmeztető üzenetet kapunk ("Az ajtó csukva van."), ha igen, akkor további ellenőrzések következnek: létezik-e egyáltalán abban az irányban bármi. Nullás célhelyiség esetén: "Nem mehetsz abba az irányba." - de lehetnek további fiktív sorszámok is, melyek például külön jelölik a falat, a sövénykerítést, a vizesárkot stb. Sőt, akár még halálos irányokat is kiképezhetünk - mondjuk, hogy egy veszélyes hegyi úton az északnyugati irány végzetes lezuhanással végződik... Csak ha minden stimmel, akkor szabad elvégezni a tényleges mozgást. (Hogy hogyan, azt is hamarosan meglátjuk.)

Egy kalandjátékban szereplő bármilyen élőlény alapvetően négyféle mozgási formát követhet a helyszínen történő kalandozása folyamán. Ezek: a közvetlen haladás, a közvetett haladás, egy távoli célpont megközelítése és az ún. "mágikus" bejárás. Közülük az első három mindegyike az öt megelőzőre épül, és egyre magasabb szintű cselekvési módokat tesznek elérhetővé.

1.) Közvetlen (vagy magyarul: "explicit") haladás esetén olyan mozgási utasításról van szó, mely pontosan meghatározza az illető által követendő haladási irányt. Az ezt kiváltó parancsok például a következőképpen hangzanak: DÉLKELET vagy MENJ BE A KUNYHÓBA stb. (Ez utóbbit az teszi közvetlenné, hogy szerepel benne a BE szócska! Máskülönben a következő csoportba tartozna - és akkor is, ha a BE irányt nem vettük volna fel az irányok közé az előző fejezetben.) Hogy ezek megvalósítása miképpen történik, azt épp az imént magyaráztam el, a továbbiakról pedig a következő fejezetben esik némi szó.

2.) A közvetett ("implicit") mozgás ezzel szemben a kalandozó részéről egy olyan kívánságot jelent, amelyben nem fogalmazza meg konkrétan, hogy melyik irányban szeretné a következő lépést megtenni - hanem csak bizonyos utalásokat tesz, az elérendő célpont megjelölésével. Ez már nehezebb feladat a programunk számára, hiszen órá háru az irány kiválasztásának felelőssége: a mondatban megadott célról el kell döntenie, melyik irányban fekszik tőlünk. Ha sikerült eldöntenie, akkor a lépés már egy egyszerű közvetlen mozgássá redukálható. Ilyenek rendkívül sokfélék lehetnek, és ráadásul valamennyiük egyéni bánásmódot igényel. HASZNÁLD AZ AJTÓT típusú felszólítás, pl. MÁSSZ KERESZTÜL A CSAPÓAJTÓN esetében az akadályok táblázatából ki kell keresni a helyszínről kivezető csapóajtót (ha van olyan), azután ennek másik helyiségét kell megkeresni a bejárás táblázat megfelelő sorában - ezzel megvan a követendő irány (ha esetleg több irány is vezet ugyanoda, a program dönthet, melyiket használja); LÉPD AT A KÜSZÖBÖT: majdnem ugyanaz, mint az előző, azzal a különbséggel, hogy miután küszöbe többféle eszköznek is lehet, ha ugyanott van mondjuk egy ajtó és egy kapu, akkor megintcsak el kell döntenünk azt is, hogy melyiket koptassuk; MÁSSZ FÁRA: egyértelműen a FEL irány helyettesítése; UGORJ: valószínűleg a LE irányt jelenti, de helyzettől függően egészen más is lehet... Főleg sorolni a példákat tovább, a dolog lényege ennyiből is

látható.

3.) Az eddigi két mozgásfajta egyszerű volt abból a szempontból, hogy mindkettő egyetlen lépés megtételére korlátozódott - valamelyik szomszédos terembe ruccanhatott át segítségükkel a játékos. Távoli célpont megközelítése esetén ugyanakkor már egy teljes útvonalat szükséges végigjárnia, méghozzá úgy, hogy ennek kiválasztását is legnagyobb mértékben a számítógépre hagyja: MENJ A KASTÉLYHOZ, KERESS EGY TISZTÁST, KÖVESD A VÁNDORT típusú parancsok ilyenek. Mint valami taxiba szálló utazó, egyszerűen közli velünk, hová óhajt elkeveredni, és attól fogva a játék idegenvezetésére bízta magát. Ez a legmagasabb szintű, legelvontabb és legnehezebben kivitelezhető haladási forma, a kalandjátékok túlnyomó többsége teljes mértékben nélkülözi. Van azért egy kicsike előnye is a közvetett mozgással szemben, az, hogy nem kell olyan sokféle eltérő kivétellel vacakolni, hanem egyetlen általános eljárásba bele lehet foglalni az egészet. Az viszont a maga nemében szép egy darab... Ismerjük ugyebár a teljes világ térképét a rajta elszórt akadályokkal egyetemben, a saját pozíciókat és azt, hogy hová kell majd eljutnunk: meg kell találnunk az egyiktől a másikhoz vezető legrövidebb (!) utat. Ha egy ember rápillant egy térképre, a megoldás szinte önmagától adódik - de egy számítógépnek ugyanezt a diót már sokkal nehezebb lesz föltörnie. Bonyolítja (vagy egyszerűsíti?) a helyzetet az is, hogy esetleg még magáról a célhelyiségről sincs pontos tudomásunk; utóvégre, ha egy játékban vagy harminc ajtó lézeng itt-ott elszórtan, és a tisztelt játékos megsúgja nekünk, hogy MENJÜNK AZ AJTÓHOZ - akkor mégis melyikre gondolt?! Nyilván arra, amelyiket leghamarább elérjük - ha legrövidebb útvonalat keresünk, akkor ez egyszersmind a legközelebb lévő. Nincs más megoldás: el kell indulni, lépésről lépésre tapogatózva előre a helyiségek végtelen óceánjában, s minden új helyszínrre lépve ellenőrizni, hogy találtunk-e a leírásnak megfelelő tárgyat. Persze nem összevissza, hanem módszeresen: sorban végigpróbáljuk mind a tizenkét irányt, s amelyik járható, arra lépünk tovább; ha végképp zsákutcába futottunk, úgy mindig visszalépünk az ezt megelőző helyre, és a következő iránnyal kísérletezünk. Ha visszaértünk a kiindulásra, akkor kudarcot vallottunk: a célpontot nem tudjuk elérni. Namost ehhez három dolog szükséges: először is maximalizálni kell a megtehető legnagyobb távolságot, mondjuk 30-40 lépésben (minden újabb lépéssel hatványozódik a lehetőségek száma! - ennyi pedig bármekkora térképhez bőségesen elég lesz); másodsor egy listában nyilván kell tartanunk az eddig megtett útvonalat (melyik helyszínről merre haladtunk tovább), hogy legyen hová visszalépnünk és tudjuk, merre kell majd mennünk; harmadszor pedig - ugyancsak e lista alapján - folyton ellenőriznünk kell, nehogy egy, már érintett helyszínrre másodsor is rálépjünk (ilyenkor úgy tekintjük az arrafelé nyíló irányt, mintha akadály zárná el - egyedül így kerülhetjük el, hogy önmagába visszatérő hurkokat írjunk le). Ezzel a módszerrel szépen meg is találjuk a keresett célpontot, csak éppenséggel egyetlen bökkenő lesz vele: korántsem biztos, hogy a legrövidebb útvonalon jutottunk el oda! Ahhoz, hogy amazt is fölleljük, kénytelenek vagyunk a keresést először egyetlen lépésre szűkítve elvégezni, s ha nem találjuk, akkor fokozatosan (mindig egy-egy lépéssel) hosszabbra engedni a pórát, minden ilyen bővítés után újra és újra végigkeresve - egészen addig, míg el nem jutunk a célba, vagy el nem érjük a bűvös határt. Lassú géppel jobb nem belekezdeni... A KÖVESD ige (egy közelben mozgó illetőnek a nyomába szegődni) a közönséges MENJ-től annyiban eltér, hogy hatósugarát már eleve csak 2-3 lépésre kell zsugorítani - azonkívül, ha sikeres volt, addig ismételni, míg az "üldözött" személy meg nem állapodik valahol. Ha igazán nagyon intelligens programmal akarjuk elkápráztatni a közönségünket, ezt az egészet esetleg még megpakolhatjuk olyan finomságokkal, hogy gyaloglás közben az emberkénk reflex-szerűen nyissa ki az útjába eső ajtókat (majd ha áthaladt rajtuk, csukja is be maga után őket...), sőt, ha zárva vannak, először próbálja végig bennük a nála levő kulcsokat, majd kopogtasson rajtuk, és így tovább...

3.) A negyedik mozgás, az ún. "mágikus bejárás" (DaCosta), nem egy

nagy szám, semmi köze az előző háromhoz, és jól hangzó nevével ellentétben mindössze azt jelenti, hogy a szokásos útvonalakat mellőzve, egycsapásra egyik helyről a másikra kerülünk. Tulajdonképp semmi hagyományos - irányokhoz kötődő - mozgás nem történik ekkor. Mindez történhet valamilyen bűbajos varázsige kimondásával, esetleg hasonló módon, mondjuk egy varázsló elteleportál bennünket máshová. Ide sorolható az az eset is, amikor valamilyen, a valóságban hosszabb műveletsort igénylő tevékenységet a játék - dramaturgiai megfontolásból - egy lépésbe sűrítve hajt végre. Például megfizetünk egy kocsist, hogy fuvarozzon el minket, s erre a program tényleges utaztatás helyett mindössze pár mondatban ecseteli számunkra hosszú utunk viszontagságait - aztán a következő lépésben már éppen kiszállunk a kocsiból a célnál. Jellegéből adódóan ezt a legkönnyebb megvalósítani.

3.1.7. Lakberendezővé változunk

A játék alapkövét letettük, világunk színterének hálózata megvan - de rajtunk kívül senki másnak nem telne benne túl sok öröme, hogy az üres helyszínek közt mászkál. A Teremtés csak félkész állapotban van: még föl kell húznunk a falakat, bevakolnunk őket, s a szobákat berendezni díszletekkel, tárgyakkal és élőlényekkel. E három dolog nem is különbözik egymástól annyira: ugyanabba a számozási rendbe bele lehet vonni az összeset, s hogy aztán melyik objektum viselkedik mozdíthatóként vagy élőként, azt már csak a konkrét programműködés dönti el, a megfelelő sorszámokra reagáló eljárásaink. Azért mindenesetre alaposan megkönnyítjük vele a saját dolgunkat, ha ezt sem végezzük teljesen logikátlanul. Tapasztalataim alapján elmondhatom, hogy úgy a legcélravezetőbb a sorszámokat megválasztani, hogy legelsőként vesszük a legmagasabb rendű, legtöbb szereppel bíró "tárgyakat": az élőlényeket.

Amiképpen az ajtók sorolásakor tettük, 1-től valameddig legyenek sorszámozva, és úgy, hogy mi magunk (azaz a főhős, a játékos!) is csupán egyik legyünk közülük, semmilyen kitüntetett szereppel ellátva - ez majd később fog jól jönni, ha azt akarjuk megvalósítani, hogy egyszerre több szereplő bőrébe is bele tudjunk bújni. Közvetlen utánuk következzenek az olyan, mozdítható tárgyak, eszközök, amelyeket föl tudunk venni és magunkkal tudunk vinni útjaink során - alighanem ezekből lesz a legtöbb. Folytassuk a feltárt a mozdíthatatlan, de azért még jelentős szerepet játszó díszletekkel (pl. egy asztal, amelyet megvizsgálva találhatunk rajta valamit, vagy szekrény, amit ki lehet nyitni stb.), utánuk azokkal, melyeknek már nincsen központi szerepük, ténylegesen csupán díszítő és hangulatfestő háttérelemek gyanánt vannak jelen a játékban (pókhálók a sarokban, fal, föld, erdő, víz...) - s a legvégére kerüljenek a rendkívül nagy példányszámban tenyésző, teljesen különleges nyilvántartású elemek (pl. ajtók). Az egyes csoportok között hagyhatunk kisebb hézagokat, kihasználatlan területeket, mert ha később netán kiderülne, hogy kifelejtettünk egy tárgyat, vagy még okvetlenül be szeretnénk szúrni valamit, akkor ne kelljen utólag átsorszámozni az egészet... Ügyeljünk ezenkívül arra is, hogy a gyűjtőnévként összevonható, hasonló funkciójú elemek (pl. a különféle kulcsok) lehetőleg egyetlen, nagy tömbben hézagok nélkül legyenek - nem mindegy ugyanis, ha a játékos egy határozatlan értékű KULCS szót használ valahol, akkor összevissza kell-e szaladgálni a különféle fõnevek között, vagy csupán az x-diktól az y-dikig terjedő területet egy rendezett ciklussal végigfutni, közelebbi meghatározás végett.

Az egyes tárgyak kezelése egy gondosan és körültekintően kialakított sorszámozással rendkívüli módon leegyszerűsödik. Pl. ha éppen egy NYISD AJTÓT utasítás végrehajtása közepette vagyunk, s már eldöntöttük, melyik ajtóról van szó, de az ajtó feltárulkoztatásához történetesen a 28-as számú bronzkulcsra van szükség, akkor annak ellenőrzése, hogy a kalandozó ezt próbálja-e használni e nemes célra, mindössze annyiból fog állni, hogy megvizsgáljuk a jelenlegi parancs eszközhatározó rekeszét: ha ez nulla, akkor kiírjuk, hogy

"Az ajtó zárva van."; ha egyenlő 28-cal (ami éppen akkor fordulhat csak elő, ha a parancshoz hozzáfűzte a BRONZKULCCSAL szót is!), akkor sikeres lesz a művelet, kinyílik az ajtó; míg ha valami más érték van ott, úgy egyszerűen annyit üzenünk neki, hogy ezt az ajtót nem nyitja az a valami, amelyet próbál. (Természetesen előtte azért meg kell bizonyosodni arról, hogy tényleg zárva van-e az az ajtó, nem pedig csak - mondjuk - csukva.) Amennyiben nagyon sok ajtónk és hozzájuk való kulcsunk van, és mindegyikük hasonló módon működik, akár ezeket is lehet egy újabb táblázatba rendezni, hogy ne kelljen külön-külön foglalkozni valamennyivel...

És pontosan erről van itt szó! A tárgysorszámot indexként (vagy keresendő adatként) alkalmazva listák és táblázatok valóságos dáridóját hozhatjuk létre, melyek a tárgyak különböző tulajdonságainak, előre megadott helyzetekben megfelelően történő viselkedésének leírását szolgálják. Jelen esetben ezek közül csak egyet említünk: a legfontosabbat, mely az objektumok térképen való elhelyezkedésének záloga. Általában ez egy olyan táblázat, amelyben a tárgysorszámot közvetlenül indexként használjuk (akárcsak a bejárési táblázatnál a helysorszámot), és valamennyi tárgyhoz tartozik benne pontosan egy-egy adat - annak a helyiségnek a sorszáma, ahol a tárgy éppen megtalálható. Egy-egy tetszőleges tárgyunk elhelyezkedése ebből szempillantás alatt kiolvasható. Ha a korábbi példáinkban felbukkant bronzkulcs teszem azt a hálószoza padlóján hever, akkor ez azt jelenti, hogy eme táblázat 28. eleme a 2-es értékre van beállítva. Ha valahol a 0-ás értékkel találkozunk, az szokás szerint azt jelenti: a hozzá tartozó tárgy "eltűnt" - semerre sem látható. Egyes díszítőelemek (pl. fal vagy padló) olyan rengeteg helyen előfordulnak egyszerre, hogy nem érdemes ezeket annyiszor külön fölvenni tárgyként, hanem csupán egyszer, s elérésük speciális módon történik: mindegyiknek saját listája van a hozzá tartozó helyekről, vagy éppen a fal esetében erre egyáltalán nincs is szükség; elvégre a bejárési táblázatban előzetesen már bevezettünk egy sajátos ál-sorszámot arra az esetre, ha valamelyik irányban fal zárna le az utunkat, így most is csak elegendő megvizsgálni, előfordul-e vajon ez az érték a pillanatnyi helyiségben valahol (s ha igen, akkor létezik itt a FAL nevű tárgy). Szintén sajátos kisebbséget alkotnak az ajtók, kapuk és más hasonló akadályok: ezek mindegyike két helyiségben található egyidejűleg, s ezekért a pozíciókért mindössze az akadálylistába kell visszanyúlni egyszer (ha ügyesek voltunk, itt is a tárgysorszámmal indexelhetünk). Nem tartozik érték ebben a táblában a gyűjtőnevekhez - azokat úgyis egy másik, konkrét tárgyra fogjuk beváltani, ha pedig nem sikerülne ez, akkor maga a gyűjtőnév-tárgy sem elérhető. Az élőlényekhez ellenben tartozik még egy másik, szintúgy roppant fontos táblázat, amelyben ugyancsak a tárgysorszámmal indexelhetünk (többek között ezért is volt érdemes a lista elejére tenni őket): az ajtókhoz hasonlóan ônekik is állapotuk van! Célzatosan a nulla érték mutassa azt, hogy valaki él, virul, egészséges és éber - az ennél magasabb számok pedig különféle rendkívüli állapotokat jelölnek (pl. alszik, elájult, részeg, megôrült vagy meghalt).

Mindebből már megválaszolásra került az az, előző fejezetben nyitva hagyott kérdés: miként mozdulunk el egyik helyiségből a másikba.

Értelemszerűen a kalandor játszóta szereplôkhöz is tartozik a táblázatnak egy olyan eleme, amelyik megmondja, hogy hol van - pusztán csak ezt kell megváltoztatni a megfelelő új értékre (amelyet a bejárési táblázatból olvastunk ki, az iránynak megfelelő oszlopban).

3.1.8. Hogyan találjunk meg valamit?

De mielőtt még hozzászoknánk a gondolathoz, hogy milyen egyszerű az életünk, szembesítsük magunkat egy megdöbbentô kérdéssel, mely az egész rendszernek hirtelen egy nagyságrenddel nagyobb mélységet és bonyolultságot ad! Mi történjék akkor, ha bármelyik tárgyunk egyben helyiség is lehet? Talán

első hallásra nem egészen világos a dolog, ezért megmagyarázom: egy reális világban gyakorlatilag tetszőleges tárgy hozzákapcsolódhat egy másikhoz, különféle módokon - például rajta áll egy asztalon, bent csücsül egy szekrényben, vagy éppenséggel a mi kezünkben van. Ilyenkor annak a tárgynak a gazdája tulajdonképpen egy helyiség, ahol a tárgy tartózkodik. De ugyanakkor a gazda-tárgy is egy másik, tágabb helyiségben létezik, ahonnan - mintegy kívülről - nézve mind a két tárgyunk látható. Ha a hordozót, mint helyiséget tekintjük, onnan nézve kifelé nemigen látunk sokmindent, viszont befelé, "mélyebbre" korlátlan bepillantást nyerhetünk. Eszerint föl kell készítenünk a programot rá, hogy mindegyik objektum hordozhassa a másikat - enélkül még csak azt sem tudnánk megoldani, hogy egyáltalán kézbe vegyünk egy kavicsot. Ezeket a tárgy-helyiségeket természetesen meg kell különböztetnünk a valódiaktól. Ha alaposabban megvizsgáljuk a kérdést, rájöhetünk, hogy bár ennek a kapcsolódásnak a valóságban tömértelen módozata elképzelhető, mégis három olyan alapvető kategóriára tudjuk leszűkíteni őket, amelyek egy kalandjáték szempontjából is nélkülözhetetlennek látszanak: vagy nála van valakinél, vagy rajta van valamin, vagy pedig benne van valamiben az illető holmi. Ha ezeket 1-től 3-ig sorszámmal látjuk el, azután 0-ás sorszámmal hozzávesszük az alapesetet, amikor közvetlenül egy helyiségben leledzik, az pontosan négyféle lehetőség. A helyiségek sorszámaikat leggazdaságosabb 2-byte-os egész számként nyilvántartani, ami így legfeljebb 65536-féle értéket vehet fel; minthogy sem tárgyból, sem helyiségből soha nincsen 16384 darab (rendszerint párszáz szokott lenni...), a 16 bit legfelső 2 bitjét nyugodtan fenntarthatjuk ennek az információnak a tárolására. A dolog ekkor úgy áll, hogy 0-tól 16383-ig számozódnak a valóságos helyiségek, előlött pedig a fiktív, tárgy-jellegű pozíciók helyezkednek el. Ha egy tárgysorszámhoz 16384-et hozzáadunk, megkapjuk azt a helyet, ahol a nála lévő valamiket keresgélni lehet (ennek csak lényeknél van értelme); 32768-at adva hozzá, a külső felülete bukkan elő; 49152-vel növelve meg, pedig a belsejébe mutatunk. Pl. ha egy zseblámpa sorszáma a 45-ös, a beléje csavart izzóé pedig 73, akkor az azt jelenti, hogy a 73. tárgy a $45 + 49152 = 49197$. helyiségben tartózkodik. Ezek az elvont helyiségek így szervesen és zavartalanul simulhatnak bele a korábban kialakított rendszerbe, egyetlen dolgot kivéve: ha valahol egy ilyen értékre bukkanunk, akkor nem elégedhetünk meg vele, mint normális helyszínnel, hanem tovább kell kutatnunk azt is, hogy az a tárgy, amihez kötődik, merrefelé található. Ennek a szervezésnek a révén számos tárgyat eldughatunk a kíváncsi játékosok orra elől, akik - mint holmi húsvéti tojásra - így csak hosszas vizsgálódás után lelnek végre rá. (Amikor megvizsgálunk valamit, a program kiírja vele együtt a hozzá kapcsolódó tárgyakat.)

Csak hogy ez így még mindig túlságosan egyszerű volna, úgyhogy azért még bonyolítunk rajta egy keveset... Utóvégre is, egyetlen ember sem egy Sámson vagy egy Herkules, és mindennek van valahol határa! Az olyannyira áhított valóságosság érdekében a tárgyak befogadóképességét korlátozni kell - egy szatyorba talán mégse lehessen belezsúfolni többet, mint amennyi befér vagy amit a fülei elbírnak. Ha valamit oda szeretnénk tenni, ahol már nincs számára hely, vagy többet próbálunk fölemelni a karjaink erejénél, figyelmeztetést kell kapnunk a játéktól, hogy nem tehetjük meg (pl. "Nem bírsz már el annyi súlyt."). De ennek a luxusnak komoly ára van: minden egyes tárgyunkhoz további három tulajdonságértéket kell bevezetnünk hozzá; ezek: a saját súlya, a teherbírása és a pillanatnyi terheltsége. Az első kettő ugyebár egy állandó érték (nem muszáj, hogy valóságos mértékegység legyen, tetszőleges viszonyzámként alkalmazható); a harmadik meg folyton változik, de nem nőhet magasabbra a másodikénál. Egy tárgy terheltségét úgy kaphatjuk meg, hogy saját súlyához az összes hozzá csatlakozó tárgy terheltségét hozzáadjuk - azért a terheltségét, és nem pedig a súlyát, mert így az az érték már összegezve tartalmazza a kapcsolódó tárgy további kapcsolódásait. Következik ebből, hogy mikor egy tárgy "üres", azaz terheletlen, akkor a terheltsége nem nulla, hanem pontosan a tulajdon súlyával egyezik meg (a

teherbírást is ennek figyelembevételével szükséges megállapítani, és lényegében csak emiatt van szükség magára a súlyra).

Többnyire egy kalandjáték leggyakoribb eljárásainak egyike egy bizonyos tárgy elérhetőségének a meghatározása: mi megadjuk neki a helyiséget,

amelyen belül vizsgálni óhajtjuk, és a keresett tárgy sorszámát - ő visszaadja azt, hogy a helyszín "felségterületén" hány darabot talált belőle. (Egynél több is lehet, hogyha gyűjtőnevet adtunk meg, vagy több ugyanolyan tárgy is szerepel a játékban.) Ha létezik ilyen tárgy, akkor azt mondjuk, hogy azon a helyszínen jelen van, vagy arról a helyről elérhető stb. Rendesen egy jelenlévő tárgyhoz kapcsolódó további tárgyak is elérhetőek szoktak lenni.

Természetesen egy szereplő kizárólag jelenlévő holmikkal cselekedhet bármit is (néhány kivételtől eltekintve, pl. amikor kérdezzük valakit egy témáról), ezért mielőtt még bármilyen parancsot is kiadni merészelne, legelső teendőnk az lesz, hogy a benne lévő összes főnév elérhetőségét ellenőrizzük. Ha pl. azt mondja nekünk, hogy HAJTSD FEL A SZŐNYEGET egy szőnyeg nélküli szobában, akkor mi keményen visszavágunk neki azzal, hogy "Nincs itt semmiféle szőnyeg." - az utasítás végrehajtásáig (vagy egyáltalán: az ige elővételéig - lehet, hogy nincs is benne ige!) még csak el sem jut a program, hanem mindenfajta cselekvés nélkül, hatástalanul továbbugrik a következő parancsra.

(Következésképpen nyugodt szívvel írhatja akár azt is, hogy EDD MEG A SZŐNYEGET, vagy csak egyszerűen SZŐNYEG - ugyanazt a feleletet kapja rá úgyszólván.) Ezzel csírájában elfojtunk mindenféle illegális tárgyra való hivatkozást, s mire a végrehajtáshoz érünk, addigra garantáltan megvan az összes főnév - tehát ezzel később már nem kell törődnünk a programban. Fontos még, hogy az is mindig jelen van, ami az általunk irányított hűshöz, mint "tárgyhoz" kapcsolódik - vagyis a nálunk lévő cuccok, rajtuk viselt ruhadarabok. Ugyanide tartozik az is, hogy a többi szereplők mind-mind féltékenyen őrzik előlünk a náluk lévő dolgokat, ergo ha egy ördögnél lévő bármilyen tárggyal próbálnánk meg babrálni valamit - kivéve, ha csupán csak megszemléljük azt -, akkor: "Az ördög nem engedi." (De ha az ördög elaludt vagy meghalt, akkor zavartalanul elvehetünk tőle bármit!)

Látjuk viszont, hogy a szőnyeg most már nemcsak a padlón, hanem akár egy asztal tetején is heverhet - mi több, még annak a tetején is állhat egy láda, s a ládában egy törpe, a kezében egy táskával, ami egy erszényt rejteget, és elképzeltető, hogy mi éppen az erszényben csöndben meglapuló aranypénzt szeretnénk kiemelni onnan... Hogy a helyzet még bonyolultabb legyen, a törpe dühösen becsaphatja az orrunk előtt a táskát, és akkor a benne lévő erszény, a pénzdarabbal együtt, eltűnik a szemünk elől; ha erre mi bosszúból rácsukjuk a törpére a láda fedelét, akkor már az egészről semmit sem látunk, csupán egy asztali szőnyeg tetején fekvő, lecsukott ládát... (De ha a ládában benne lennénk mi is, akkor megintcsak látnánk a törpét, eltekintve attól, hogy odabent sötét van.) Magyarul, a tárgyak tetszőleges mélységig egymásba ágyazódhatnak, s hogyha benne van egy tárgyban valami, akkor még azt is figyelniük kell közben, hogy nyitva van-e a hordozó vagy csukva. Ahhoz, hogy a pénzérme elérhetőségét megtudjuk, először is az erszény elérhetőségét kellene ismernünk, ami viszont a táskától függ, és így tovább - ezt a keresést tehát egyedül egy rekurzív eljárással tudjuk megoldani, olyannal, mely minden egymásba ágyazódás esetén újra és újra önmagát hívja meg. Ha eközben bármelyik szinten negatív válasszal tér vissza, akkor az végigvonul a teljes láncon, és az eredményünk elutasító lesz; ha nagynehezen mégiscsak elvergődünk a gyökéremig, azaz egy valóságos helyiségig (és az megegyezik a keresett helyiséggel!), akkor megvan a tárgy, és elérhető.

Az igazsághoz hozzátartozik, hogy ennek az egész tárgy-kezelésnek van egy elég komoly korlátja. (Nem a rekurzív keresésre gondolok, hanem az előző fejezetben kifejtett táblázatos nyilvántartásra!) Csak akkor alkalmazható hatékonyan, ha mindegyik tárgyból egy vagy több - de konkrétan előre meghatározott számú darab létezik. Pl. egy fáról leszedünk egy almát - az egy darab. Leszedünk egy másikat - most már kettő van a kezünkben. De ha leszedünk

százat, akkor száznak kéne lennie - ami viszont ebben a rendszerben kizárólag úgy oldható meg, ha már eleve, a játék indításakor is létezik valahol mind a száz alma, pontos, külön sorszámokkal ellátva. Ha azt szeretnénk, hogy bármelyik tárgyból lehessen hasonlóan száz darab, akkor már eleve mindegyikből legalább ennyit kellene a játékba beleterveznünk. Párszáz tárgy helyett így lenne több tízezer - iszonyatos pazarlás a tárral és az idővel! Egyszóval, ez a kezelési rendszer, bár egyszerű, de túlságosan merev. Az esetek túlnyomó többségében ugyan tökéletesen beválik, de ha valamivel igényesebbek vagyunk, akkor meg kell próbálnunk ugyanezt valahogy dinamikusabban megvalósítani. Erre megoldás lehet, hogyha egyetlen, statikus táblázat helyett egy amolyan változó hosszúságú puffer-táblázatot készítünk. Ennek minden eleméhez jóval több adat tartozna, mint eddig: lenne egy, mely megadja, hogy milyen tárgyról van szó (tehát az eddig használt tárgysorszám), s egy másik, mely azt jelölné, hány darab van belőle. A harmadik volna a helyszín, amelyen megtalálható - ez annyiban módosulna, hogy a fiktív tárgy-helyiségek alapját most már nem a tárgysorszám jelentené, hanem egy, a pufferon belüli másik elemre mutatna tovább. (Negyedikként pedig hozzájönne még az aktuális terheltsége - a súly és a teherbírás elég, ha továbbra is a tárgysorszámhoz kapcsolódóan van meg, ugyanis azok állandó értékek. Ellenben ne felejtsük, hogy az összeadandókat itt még a darabszámmal is szorozni kell!) Ha új tárgy bukkanna elő valahonnét, először mindig helyet kéne neki foglalni a pufferban, s hogyha betelne az egész terület, akkor lehetne sorjában eldobálni a régebbieket. A száz alma így mindössze egyetlen bejegyzést foglalna el, amiben 100 volna a darabszám (amikor további almákat gyűjtünk a már meglévők mellé, azokat egyszerűen hozzá kellene adni ehhez). Ha kettesével szétpakolnánk őket ötven különböző helyiségbe, akkor már ötven bejegyzésünk lenne, 2-es darabszámokkal. Amikor keresünk egy tárgyat, a közvetlen indexelés helyett végig kellene olvasnunk a puffer teljes méretét, vizsgálva, hogy melyik elemben egyezik a tárgysorszám a kívánttal; amikor egy tárgyat törölünk, az az összes ôhöz kapcsolódót is magával rántaná. Vigyázni kéne, hogy a fontosabb tárgyak ne tűnjenek el.

3.1.9. "Sokasodjatok és növekedjete!"

Az eddigiek folyamán egy meglehetősen kihalt és sivár valóságot építettünk föl - egyelőre úgy fest az egész, mint az ôstengerek, a bennük úszkáló egysejtűek nélkül. Hiányzik belőle valami, ami elevenné teszi azt: a játék szereplői még nincsenek bekapcsolva az események vérkeringésébe. Az élőlényeket nemcsak az különbözteti meg a többi tárgytól, hogy a sorszámozás letelezésén foglalnak helyet, hanem hogy sokkal változatosabb módokon lehetséges érintkezésbe kerülni velük, illetve a mi közreműködésünk nélkül is mindenféle "akciókra" ragadtatják magukat. Szélsőséges esetben ezek akár valóságos személyek is lehetnek - ld. a második fejezetben megemlélt hálózatos kalandjátékokat -, máskülönben nekünk kell a program segítségével emulálnunk őket. Az egyes élőlényeket leginkább az különbözteti meg egymástól, hogy különféle közeledési kísérleteinkre eltérő módon reagálnak. Vannak bizonyos alapvető magatartás-formák, amelyeket egy jó kalandjátéknak föl szükséges ismernie, és általános jelleggel lekezelnie őket: agresszív cselekedetek (megütni vagy megtámadni valakit), a beszélgetés különböző formái (köszönni, beszédbe vegyülni általában, vagy pontosan meghatározott dolgokat mondani, esetleg faggatózni jól körülírt témákról) és a tárgy-csere jellegű dolgok (megmutatni, odaadni vagy elkérni egy tárgyat, netalántán pénzt). Ezeket valamennyi szereplőre nézve külön-külön ki kell dolgozni, úgy, hogy látszólag egyéni módon reagáljanak.

Sajnos, a legtöbb kalandjáték úgy készül, hogy már eleve egy szilárdan lebetonozott nézőpontból mutatja a benne zajló eseményeket. Ez azt jelenti, hogy létezik egy e célra kiválasztott főhős, és mi mindent egyedül az ő szemén keresztül látunk. Kétségtelenül ez a legkényelmesebb megoldás, ám

sokkal izgalmasabb a játék, hogyha kívülről is megfigyeljük benne önmagunkat közben. Programozástechnikailag legjobb, ha - a manapság divatos objektumorientált programozáshoz hasonlóan - a játék szereplőinek mindegyike egy önálló, zárt egységet alkot, s valamennyiükhöz tartozik egy-egy, a magatartásukat generáló program vagy programrész, amiről a vezérlő főprogram gyakorlatilag semmit nem tud, csupán csak az általuk jelzett cselekvési szándékaikat látja. Ebben a felállásban - a főprogram szempontjából legalábbis - maga a játékos, "a" főhős is csak "egy a sok közül", a többiekkel egyenrangú lény, aki hasonló korlátokkal és lehetőségekkel rendelkezik, akárcsak balsorsában osztozó társai; más kérdés, hogy - valódi léténél fogva - ezekkel a lehetőségekkel hasonlíthatatlanul sokoldalúbban képes élni és cselekedni, mint ők. Ez egyúttal azt is jelenti, hogy bármilyen lépésre képesek vagyunk, azt akárki más is megtehetné helyettünk ugyanebben a helyzetben - elméletileg -, más szóval a többieket is aktívan bevonhatjuk a játék feladványainak teljesítésébe; ahelyett, hogy magunk tennénk meg valamit, megkérünk rá valaki mást. (Pl. NYISSA KI AZ AJTÓT helyett így: KÉRD MEG AZ ŐRT, HOGY NYISSA KI AZ AJTÓT.) Ettől egyrészt mindjárt változatosabb lehetőségeink nyílnak, hiszen ha többször próbálkozunk valamivel, nem kell mindig ugyanazon a módon megfogalmaznunk, amit szeretnénk; másrészt pedig meglehet, vagy a program netán el is várja tőlünk, hogy másokat is megmozgassunk a végső siker érdekében: lehetnek benne olyan epizódok, amelyek feloldása kizárólag csapatmunka árán lehetséges. Például áll valahol egy őrszem, aki egy bizonyos ponton nem enged átlépni senkit - de ha egy szereplőnkkel ügyesen eltereljük a figyelmét a feladatáról, akkor másvalaki észrevétlenül elsurranhat a háta mögött közben... Egycsapásra érdekesebbé válnak ily módon a játékunk feladványai! Arra is lehetőséget nyújt ez a rendszer, hogy akár egyszerre több szereplő bőrébe is belebújjunk, állandóan váltogatva azt, hogy melyiküket alakítjuk éppen. Egy játék résztvevőinek viselkedése az alábbi forrásokból származhat:

1.) Közvetlenül a billentyűzetről begépelte utasítások határozzák meg számára a következő lépést. Tipikusan a főszereplő magatartása ilyen. Egy főhős személyének kiválasztása nem is olyan bonyolult, mint hinnénk: adott a szereplők listája, amelyek közül választani lehet, s mindössze egy változóban folyton nyilván kell tartani annak sorszámát, akik jelen pillanatban éppen vagyunk. Amikor a játékos azt mondja, hogy ÉN, MAGAMAT, ENGEM... stb., ezeket a személyes névmásokat kicseréljük az itt tárolt sorszámra, üzenetek kiírása közben pedig annak neve helyett a "te" (Ön), "magad" (maga)... stb. alakokat kell - persze helyesen felragozott formában - használni. Na ez utóbbival szokott inkább gond lenni - szeretjük az üzeneteket kész szöveggé, rögzítetten tárolni, és nemigen fűlik a fogunk hozzá, hogy mindenféle képlékeny kifejezésekkel bonyolítsuk őket. Elvégre ez nemcsak egyetlen szónak a cseréjét jelenti, de a teljes szövegkörnyezetet is hozzá kell igazítani ehhez. (Pl. "Valaki kinyitja az ajtót." helyett "Kinyitod az ajtót." - az alany mellett az állítmány is megváltozik stb.) Ha azt akarjuk elérni, hogy egyes tevékenységekre csak bizonyos "kiválasztottak" legyenek képesek, nem kell mást tenni, mint előtte ellenőrizni, hogy éppen ők vagyunk-e mi.

2.) Ösztönös sugallatait egy külön e célra készített, irányító programrészről kapja. Ez kétféle módon történhet: vagy adott helyzetben egy előre lerögzített cselekménysort hajt végre (pl. van valahol egy éjjeliőr, aki bizonyos időközönként egy határozott útvonalat végigjár, és ellenőrzi, minden rendben van-e arra - napközben meg lefekszik aludni, és ki lehet figyelni az őrségváltás idejét stb.), avagy pedig részben véletlenszerűen állítja elő ezeket a program (egy-egy alakok a játékban szeszélyesen kóborolnak ide-oda, miközben időnként az "eszükbe jut" valami - mondjuk becsukni egy ajtót vagy eldobni egy tárgyat stb.). Esetenként ez az előbbi ponttal kombinálódhat, olyképpen, hogy a saját figuránkon keresztül megkérünk rá valaki mást, hogy tegyen meg számunkra valamit (az említett KÉRD MEG AZ ŐRT... kezdetű példa!), s ha jó kedvében találjuk a fickót, akkor lehet, hogy még engedelmeskedikis... Nem könnyű - de igen hálás! - feladat minden egyes szereplőnkhez különböző programokat írni, és bizony komoly előzetes tervezést igényel összhangba hozni számos olyan eseményt, amelyek egyidőben, de a játéktér különböző pontjain mennek majd végbe...

3.) Valahonnan "kívülről", a számítógép számára ismeretlen külvilágból érkeznek a parancsai - rendszerint már előértelmezett, azaz a legrövidebb, legtömörebb alakra faragott formában. Ez az eset akkor állhat elő, amikor egy olyan programot írunk, amivel egyszerre több felhasználó is üzekedhet - az egymással összeköttetésben álló felek a hálózaton keresztül kommunikálnak egymással. Vegyük észre, hogy ez alig különbözik az első ponttól, szinte csak annyiban, hogy itt egyszerre több kalandorhoz kell hozzárendelt sorszámokat nyilvántartani.

Ezenkívül gondoskodni kell még arról, hogyha egy szereplő valamilyen módon "önkívületi" állapotba került (meghalt vagy elájult), akkor az eredeti irányítás ne érvényesüljön nála, és ne is kommunikálhasson a környezetével többet (de bizonyos idő elteltével azért újra magához térjen - kivéve, ha meghalt). Az eddigiek konkrét megvalósítása nehéz, de nem túlságosan bonyolult programozói feladatot jelent (meg lehet írni egy általános szervező programot, ami minden lehetséges bemenetet lekezel), sokkal keményebb mogyoró azonban ennek a fordítottja, vagyis az egyes résztvevők tájékoztatása a különféle történések hatásairól. A gyakorlatban ez annyit tesz, hogy a központi résznek a játék világában végbemenő minden egyes eseményt figyelnie kell, és erről az összes szereplő felé valamilyen üzenetet továbbítania - mégpedig minden egyes szereplő felé más-más tartalmú üzenetet! Ráadásul az egyes események is - mintegy következmény gyanánt - további eseményeket vonhatnak maguk után. Egy nagyon egyszerű példával illusztrálva a dolgot: ha valahol egy Tompika nevű szereplő pl. kinyit egy ajtót (hogy továbbra is ennél az oly sűrűn fölbukkanó motívumnál maradjunk), akkor nem elég az ajtó állapotát nyitottra változtatni, hanem az illetőt értesíteni is kell róla: "Kinyitod az ajtót." (Az egyszereplős kalandjátékok meg is elégednek ennyivel!) A vele azonos helyszínen tartózkodókat ugyanakkor már arról kell tájékoztatni, hogy "Tompika kinyitja az ajtót." - míg végül a távol maradóknak abszolúte semmit nem szabad megszimatolniuk erről az egész cselekményről. Sőt, még egy negyedik csoport is van: az ajtó túloldalán állók nem tudhatják, ki cselekszik, így mindössze annyit észlelnek belőle, hogy "Valaki kinyitja az ajtót." Ezenkívül a helyzetet még tovább bonyolíthatja, ha teszmez a kétféle helyiség közül valamelyikben sötét van: ekkor tudniillik az ott-tartózkodók nem láthatják, amidőn Tompika kinyitja az ajtót, tehát akkor ezeket nem is szabad értesíteni róla... Viszont megeshet, hogy a másik szobából, az immáron nyitott ajtón keresztül beárad a fény, aminek következtében ezek a bácsik nemcsak hogy az ajtó kinyitásáról szereznek tudomást, de ráadásul még megpillantják azokat a dolgokat is, amiket eddig a teremben honoló sötétség eltakart a szemük elől. (Tehát mindjárt a helyszínelírást is ki kell nekik írni, és az egész szoba kezelése megváltozik.) És akkor akár ez még folytatódhat azzal, hogy mondjuk a hirtelen világoosságban valaki fölismeri az ellenségét, és azonnal rátámad... Vagy nála van egy tekercs film, ami fényt kap és tönkremegy stb. Egy ilyen egyszerű döntés is, mint amilyen egy ajtónak a kinyitása, alkalmasint a történések egész láncolatával (vagy inkább: hálójával) járhat együtt - amiről pedig szintén mindig tudósítanunk kell az érintett felek mindegyikét. Egy dinamikus szerveződő kalandjátékban lépten-nyomon előfordulhatnak hasonló jellegű problémák, ahol rengeteg, egymástól látszólag független külső körülmény együttes hatását kell figyelembe venni.

3.1.10. Az idő kerekéhez kötve

Egyáltalában nem mindegy, hogy egy kalandjátékban miféle ütemhez igazítva telik-múlik az idő. A legegyszerűbb megoldás, különösen az egyszereplős kalandjátékokra leginkább jellemző, hogy mindig egy-egy sikeres parancs végrehajtása lépteti tovább az egész rendszert egyetlen időegységgel, mintha valamiféle, változékony hosszúságú órajelet adna neki ezzel. Ameddig gépelünk, a következő mondatunkat szerkesztjük, addig a játékban áll az idő -

mihelyst azonban leütjük az ENTER-t (és be is írtunk neki valamit), egycsapásra mindenki öregebbé válik, mondjuk egy negyed órával. Tulajdonképpen nem is időhöz, hanem lépésszámhoz vannak szinkronizálva egy ilyen játék történései - pl. olyan időzítéseket alkalmaznak bennük, hogy - mondjuk egy kopogtatást követően - öt lépés múlva kinyitja nekünk az ajtót valaki. Tetszőleges ideig lehet töprengeni a pillanatnyi helyzet megoldása fölött, és hiába ugrik a nyakunkba egy vérszomjas démon, nyugodtan elmegetünk megvacsorázni, mielőtt végképp az arcunkra fagyyna az az utolsó vigyor... A program belső számlálót, amelyek az eljövendő események titkos előhírnökei, a végrehajtott utasítások csökkentik rendszeresen eggyel. És ez nagyon helyesen van így! Egy elsősorban képzeletre és gondolkodásra épülő játéknál ez egy ideális helyzet - tökéletesen megengedhető és elfogadható, hogy ki-ki a saját belső ritmusa szerint haladjon előre a feladatok megoldásában. (Hogy is nézne ki, ha egy gyakorlott gyors- és gépíró illetéktelen előnyre tenne szert a lassúkezűvel szemben...)

Csak hogy egy többszereplős kalandjátékban már egyszerűen tarthatatlanná válik ez az állapot. Ha a program mondjuk hat különböző játékos parancsait fogadja párhuzamosan, akik mind eltérő ritmus szerint gépelnek, akkor mégis melyikükhöz alkalmazkodjon a többi? Amikor az egyik semmit nem csinál, hanem csak karbatett kézzel és összeráncolt homlokkal bámulja a képernyőt, akkor is kötelessége a programnak, hogy pontosan a történések idejében haladéktalanul értesítse őt a másik lépéseiről, ha az éppen akkor halad el mellette, vagy ugyanazon a helyszínen tevékenykedik. Az efféle programokban eszerint nem tehetünk mást: valós idejű időzítéseket vagyunk kénytelenek alkalmazni bennük. Ez viszont maga után vonja, hogy a bevitel és kiírás funkcióját az eddigieknél sokkal élesebben el kell határolnunk egymástól - mert mi történjék, hogyha véletlenül éppen akkor érkezik valakihez egy fontos kiírandó üzenet, amikor ő saját mondata szerkesztésének a kellős közepénél tart; esetleg föl sem pillantva a képernyőre, keresi a megfelelő billentyűt? Nem lehet csak úgy otrombán kettévágni a félig begépelte mondatot, és a közepébe belenyomtatni a szöveget! (De lehet: a TELNET-es játékok sajnos pontosan ezt teszik...) Okvetlenül kétfelé kell bontanunk a képernyőt, úgy, hogy külön legyen egy felület a kiírás és szintén külön a bevitel számára (utóbbinak két-három sor is elég lesz). Így azok már nem zavarják egymást - vagy mégis?! Mit tehetünk akkor, ha a megjelenítés mezejében egy különösen hosszú leírás kezd el szépen, megfontoltan kibontakozni az ismeretlenség homályából, mialatt mi szorgosan a parancsunkat gépeljük? Bevett gyakorlat, hogyha egy szöveg hosszabb annál, mint amennyi a képernyőre egy adagban kifér, akkor oldalanként meg-megszakítva, minden oldal végén egy billentyű-lenyomásra várakozva fokozatosan léptetjük azt tovább (vagy magyarul funkció). De még ha ez az eset nem is forog fenn, akkor is az ablak görgetése több másodpercig is eltarthat, és rendkívül illúzióromboló lenne, ha erre az időre hirtelen megakadna az alsó sorokban a bevitel. Másik probléma: mi van, ha a kiírás-ablakban egy félig megjelenített szöveg éppen ENTER-rel való továbbléptetésre vár, de mi nem törődünk vele, és zavartalanul csak a mondatunkra figyelünk - hol várakozzon addig a szöveg hátralevő része, és ha kiírás közben újabb üzenetek érkeznek, azokat miképpen várakoztassuk? Meg kell oldanunk tehát azt is, hogy a kétféle funkció ne csak térben, de időben is egymástól teljesen független és párhuzamos legyen: mialatt gépelünk, tényleg aközben folyjon odaát a kiírás! A még kiíratlan, de a küldőtől már átvett "szűz" szöveget pedig addig is egy átmeneti pufferban kell tárolnunk, ahol egy bizonyos határig gyűlhetnek és halmozódhatnak a sorok és a mondatok, de ha a puffer betelt, akkor haladéktalanul ki kell görgetnünk őket a képernyőre - akár tetszik a felhasználónak, akár nem. Különösen mókás tud lenni, amikor a szereplőt a játékban egy súlyos baleset érte, de ő még valahol tíz oldallal följebb tart a szövegek olvasásában, miközben a többiek már réges-régen értesültek róla, hogy ájultan hever a földön, és apránként kipakolják a hátizsákjából az értékesebb cuccokat...

Egyetlen megoldás létezik a legsimább párhuzamosság elérésére, az, ha a program központi, vezérlő része sohasem "ragad le" valamilyen szubrutinnál, hanem egy örökös végtelen ciklusban megállás nélkül kering három alapvető tevékenység: a beviteli mező szerkesztése, az események végrehajtása (beleértve természetesen a saját és a többiek által kiadott utasítások végrehajtásait is) és a szövegkiírás alapvető fázisai közt. Ehhez az szükséges, hogy valamennyi funkciót apró, szétválasztható és önállóan végrehajtható kis lépésekre tagoltan valósítsuk meg. A szövegkiírás esetében

ilyen építőkocka lehet pl. az átmeneti puffer egyetlen sorának kiléptetése a képernyőablakba, vagy a beviteli rutin esetében egyetlen lenyomott billentyű beolvasása a billentyűzet-pufferből (ha van olyan), és annak megfelelően a beviteli mező módosítása. A program tehát úgy fog működni, hogy folyton figyel, történnie kell-e valamilyen eseménynek, s ha igen, akkor végrehajtja azt, és a neki megfelelő szöveget a puffer alsó végéhez hozzácspja (ez egyetlen szemvillanás alatt megvan); majd rögtön továbbadja a vezérlést a kiíró rutinak, amelyek a szövegpuffer legfelső sorát kigörgeti a képre (hacsak nem üres a puffer éppen akkor); ezt követően a beviteli szubrutin lép színre, mely pedig egyetlen karaktert beolvas, ha képes; végezetül pedig vissza az elejére, és ez így megy tovább megállás nélkül... Nem kis feladat elérni, hogy ez az egész így egyben kellőképp összehangoltan fusson - különösen, hogyha grafikus képernyőt használunk a megjelenítéshez, ami már eleve alaposan lelassítja a kiírásokat. (Ajánlott az ASSEMBLY nyelv használata például.)

Többek között emiatt is számít, hogy mind az értelmező szubrutin, mind pedig a program más egyéb végrehajtó eljárásai igen-igen serényen végezzék a dolgukat - mert miért ne fordulhatna elő, hogy húsz különböző játékos egyszerre adja ki, egyenként hatszáz betűből és harminc parancsból álló mondatait? Amiből mindjárt egy másik szempont is következik, nevezetesen hogy az egyes játékosok teendőit is hasonló pufferokban kell előzetesen nyilvántartani - elvégre mindnyájan kiadhatnak több parancsból álló mondatokat is, amiből egyelőre még csak az első vagy a második hajtódik végre, amelyeket addig is tárolni kell valahol, de ő gonosz mosollyal az ajkán tüstént begépeli máris a következő adagot, és lehet, hogy ezt egyszerre akár többen is megteszik...

És ne gondoljuk azt sem, hogy mindezek a szörnyűségek kizárólag hálózatos kalandjátékokban fordulhatnak elő! Ha egy olyan kalandot készítünk, amiben ugyanazt az egyetlen számítógépet használó játékos több szereplőt is irányíthat egymással párhuzamosan, mondjuk ESC-pel vagy TAB-bal kapcsolgatva az egyik vagy a másik között (esetleg osztott képernyőn megjelenítve egymás mellett egyidejűleg akár többet is), miközben valós idejű megjelenítést alkalmaztunk benne, az pontosan ugyanakkora galibákat teremthet, mintha ezren küldözgetnék az utasításaikat a komputernek egyszerre! De megéri a belefektetett munkát a dolog, mert rendkívül látványos lesz a végeredmény. Ezáltal tehát elértük, hogy - elméletileg - akárhány játékos ténykedését le tudjuk kezelni egyidejűleg, miközben sem a végrehajtás, sem a folyamatosan a háttérben zajló szövegkiírás nem zavarja egymást, és főképpen pedig a bevitel mindeközben zökkenőmentes marad. De nyitva maradt még egy probléma: a különböző játékosok különböző sebessége. Ha valaki gépír, mint a villám, annak a parancsai is fokozott ütemben hajtódnak végre, amire még ráfoghatjuk, hogy megérdemli ezt a kis előnyt, még ha a többiek nemhogy reagálni rá, de még csak figyelemmel követni se nagyon bírják közben; de ha egy több parancsból álló mondatot írt be valaki - erre ugyebár egy lassúbb illető is képes -, akkor már igazán tisztességtelenül manőverező vágtazásba kezdenek az utasításai - ennek egymást követő lépéseit a program mindenfajta késleltetés nélkül, teljesen egybefolyva, mondhatni szinte egyszerre vágja a többiek képébe. Fokozott mértékben áll ez a számítógép által irányított szereplőkre (ún. NPC-k: ez a "Non Player Character" - "nem játékos szereplő" -

angol nyelvű kifejezés rövidítése): önük tudniillik abszolúte semmiféle gépirásról nincsen szó - az illető fickó cselekvési szándékai a másodperc törtrésze alatt alakulnak ki egy külön e célra berendezett "műhelyben". Ennek eredményeképpen aztán ezek olyan követhetetlen ámokfutásba kezdenek, hogy szinte látni se nagyon fogják őket a többiek, amint nagyritkán föltűnedeznek egy-két tizedmásodpercre itt-ott... A megoldás kézenfekvő: minden egyes élőlénynek - függetlenül attól, valódi-e vagy NPC - osszunk ki valamilyen maximális sebességértéket, amivel haladhat. Ez pl. úgy néz ki, hogy mindenkihez tartozik egy-egy számláló, melyeket bizonyos időközönként csökkentünk, s ha nullára csökkent, akkor következik lépésre az az illető. Ezeket egytől néhány másodpercig terjedő tartamokra célszerű beállítani, miáltal mindenkinek személyre szabott sebessége lehet - bizonyos játékosokat vagy NPC-eket tetszőlegesen fölgyorsíthatunk vagy lelassíthatunk, amivel a helyzetek nehézségét is lehet némiképpen állítani. Az említett értékeket akár dinamikusán is változtathatjuk: pl. minél jobban meg van pakolva súlyos tárgyakkal valaki vagy minél fáradtabb és kimerültebb, annál jobban lelassul a mozgása stb.

3.1.11. Néhány jótanács és további lehetőségek

Az eddigiek során eléggé nagyvonalúan és magas szinten kezeltük a témát - ha ugyanezt részletekbe menően kéne kifejteni, egy egész könyvet sem volna nehéz megtölteni a kalandjáték készítéséről. (F. DaCosta megtette már ezt, de ő csak az egészsnek az alapjaival foglalkozott.) Most befejezésül mindössze néhány hasznos jótanácsot szeretnék felsorolni azok számára, akik komolyabban szeretnének foglalkozni ilyesmivel, illetve egy kicsit eltöprengeni rajta, hogy mire is lehetne még fölhasználni a leszűrt tapasztalatokat.

Először is, a kalandjátékok világában kezdettől fogva léteznek bizonyos kialakult szokások és elvárások, amiknek nem árt, ha az újdonsült jövevények is megpróbálnak eleget tenni. Nem elég egy helyiségekből álló térképrendszert fölépíteni, és megtölteni tárgyakkal és élőlényekkel - ahhoz, hogy ez igazán kézzelfoghatóan megjelenjen a játékosok számára, az is szükséges, hogy az információknak lehetőleg minél bőségesebb tömegével elhalmozzuk őket azok milyenségét illetően. Mint ahogy a való világban, itt is a szemét használja leggyakrabban, azaz mindent, ami csak elérhető, vizsgálni és tanulmányozni igyekszik: kétségtelenül a mozgási parancsok mellett a leggyakrabban alkalmazott funkciójuk a VIZSGÁLD ige. Úgy kell megoldanunk, hogy ezzel minél többmindent elérjen, és lehetőleg minden tárgyról, amelyet megvizsgál, hosszú és részletes leírásokat bocsássunk a rendelkezésére. Ha mondjuk egy helyszínnek a leírásában hangsúlyozottan szerepel, hogy "az erdőben, egy hatalmas fenyőfa tövénél állsz", akkor nem túlzottan szerencsés, ha egy NÉZD MEG A FENYŐFÁT próbálkozásra valami ilyesmi lesz a válasz: "Nem értem azt, hogy fenyőfát." - vagy: "Nincs itt semmiféle fenyőfa." Ilyesmi a legjobb körökben is előfordul néha, elvégre mindent belezsúfolni még a legbővebb memóriába sem lehetséges, de törekednünk kell rá, hogy minél ritkábban forduljon elő. Ha pedig már fölvettük a szótárba, és vizsgálni is engedi a program, akkor ne csak valami olyasmi kétszavas közhelyet vessünk oda neki, hogy "Szép nagy fa." - hanem aprólékosan írjuk le, ahogy a kérge pikkelyei repedésekkel tarkítva borítják a törzsét, amelyet egyébként tömény, bódító illat vesz körül, földből kiálló gyökerei körül arasznyi vastagon fedik a talajt a többéves, bomló fenyőtüskék, itt-ott kibuggyant belőle a ragacsos gyanta, és egészen elszedülünk, amikor a magasban eredő legalsó ágaira nézünk... Ennek nemcsak öncélú szórakoztató szerepe van a játékban, hanem így burkoltan utalunk esetleges további tárgyra, melyeket a játékos észrevesz, ha egy kicsit is kísérletező hajlamú. Pl. az említett avart fölpiszkálva egy odahullott fenyőtobozt találhat, amire majd a játék valamelyik másik pontján

lesz szüksége, vagy a törzsről lekapart gyantát ragasztóként alkalmazhatja máshol stb. A lényeg az, hogy nem szabad mindent direktben az orra elé tájni, hanem el kell rejteni az értékesebb dolgokat, és a játékosra bízni, hogy apránként elmélyedve fölfedezze őket. Láttuk, miképpen lehet a tárgyakat egymáshoz kapcsolni és egymásba ágyazni, így most már nem a szoba közepére helyezünk majd mindent egyetlen halomban, hanem szétszétva mindenféle asztalokkal, polcokkal és szekrényekkel töltjük meg a falakat, melyek mindegyike akár egy egész külön kis világot is tartalmazhat elrejtve magában. Ez a helyiségek elrendezésére is vonatkozik, azaz ne egyetlen kijelölt útvonalon lehessen végighaladni rajtuk, sőt, még csak egyáltalán ilyen "útvonalak" se létezzenek a játékban, hanem olyan legyen az egész, mint egy nagy, kerek, zezugos arborétum, váratlanul egymásba visszakanyarodó ösvényekkel, és egyedül a játékoson múljon, hogy merrefelé mozog rajtuk - ne akarjuk a szájába rágni semmilyen előzetes elképzelésünket. (Vagy ha igen, akkor azt úgy tegyük, hogy ő ezt ne vegye észre...) A játékban éppenhogy az elrejtett, nem látható dolgok jelentik a legfőbb vonzerőt - ha tudjuk, hogy ott vannak, és mégse találjuk meg őket...

Nagyon sokan elrontják a játékot azzal, hogy mindenáron valamilyen történetet akarnak elmesélni általa; holott egy kalandjáték egyáltalán nem erről szól! Az persze nem baj, ha van ilyen - sőt, egy jól kitalált kerettörténet sokat emel a játék színvonalán -, de vegyük tudomásul, hogy ez egy másodlagos dolog: kicsit szigorúan azt is mondhatnánk, hogy a bevezetőnél tovább nem szabad(na) menni ezzel. Miért? Ha mi elejétől a végéig kimódoljuk a cselekményt, majd a szereplőket arra kényszerítjük, hogy lépésről lépésre ezen az úton haladjanak végig, azzal elveszünk tőlük azt az illúziót, amiért voltaképpen leültek a számítógép elé: hogy a játék tőlük függ, hogy szabadon ők alakítják az eseményeket. Ha így teszünk, azzal nyíltan uralkodni akarunk a főhősön, amivel bizony könnyen elvehetjük a kedvét a játéktól - az ilyesmi csakis ellenérzéseket szülhet. Az efféle túlszerkesztést leggyakrabban attól való félelem motiválja, hogy a játékos esetleg átsiklik a kedvenc ötleteink fölött, egyszerűen nem veszi észre azt, amit mi a legnagyobb poénnak szántunk - s ennek elkerülése végett mindenáron megpróbáljuk bebiztosítani a dolgot, mintegy szájbarágni a következő lépést. És ezzel - akaratlanul is - agyoncsapjuk az egészséget: kapkodva lerántjuk a leplet az összes titokról, míg végül szegény kalandor ott áll teljesen kiábrándultan. Egy kalandjáték tartalmát nem egy történetnek az elmesélése adja, hanem hogy tulajdon képzeletünkből merítve fölépítünk egy miniatúr világot, ami a megfelelő eszközökkel azt a látszatot kelti magáról, hogy él - s ennek a célnak kimondottan jól tesz, ha az őt alkotó részletek és epizódok lazán szétszórtan hevernek, és minél kevesebb összefüggést mutatnak egymással a felszínen. (Hosszú ideig eltart, mire az ember megsejti a látszólagos káosz mögött mélyen meghúzódó rendet.) A jól sikerült játék nem egy vagy több, előre elrendelt útvonalat jelent, hanem úgyszólván egyszerre terjeszkedik valamennyi irányban. Minden lehetséges eszközzel akadályozni kell, hogy világunkról pontos, részletes térkép készüljön! Álljon minél nagyobb számú helyiségből az egész, és azok is minél bonyolultabban kapcsolódjanak egymáshoz (pl. ne csak vízszintesen, hanem függőlegesen is keresztül-kasul, akár egy többszintes barlangrendszer esetében) - úgyhogy ha valami vakmerő nekiállna ábrázolni őket, hát minimum A1-es papírra legyen hozzá szüksége... Minél több változó körülmény legyen! (Pl.: sötét helyek, ahol világítani kell, de a lámpában egy idő múlva kimerül az elem; véletlen utakon bolyongó, csavargó szereplők; pénz, amivel vásárolni lehet, de igencsak szűkében mérjük számára; vízalatti zugok, melyekben csak rövid ideig tartózkodhatunk megfulladás nélkül; napszakok váltakozása - éjszakára bezárnak a boltok, becsukják a városkaput - stb.) Szélsőséges esetben akár még az is elképzelhető, hogy napszakok szerint változik a helyszínek leírása, úgyhogy ha éjszaka járunk ugyanazon a helyen, akkor meglehet, egészen más dolgokat találunk ott, mint nappal... Csak részletesebb vizsgálódás árán fölfedezhető rejtetekutak kellene (festmény

mögötti rejtekajtó, csapóajtó az ágy alatt, létra a kútban...), lelakatolt ajtók, egyirányú átjárók, helyüket változtató ôrszemek. Sok-sok csukott ajtó legyen, és más hasonló akadályok, s még az sem kizárt, hogy némelyik ajtót netán sohasem lehet kinyitni, így a kalandor örökké csak találgathat, mi is lehet mögötte. Roppantul lényeges, hogy minél több olyan epizód szerepeljen benne, melynek nemcsak egyféle megoldása van! Ha mondjuk be szeretnénk jutni egy fallal körülkerített városba, akkor cselekedhetünk úgy is, hogy bizonyos összeget fizetünk a főkapu ôrének, aki erre kinyitja nekünk a főkaput; esetleg ha elég erôsek vagyunk, megölhetjük és elvehetjük tôle a kulcsokat; valahol, egy másik ponton átmászhatunk a kôfalon; vagy akár egy földalatti alagúton keresztül is behatolhatunk, hogy aztán az egyik lakóház pincéjében bukkanjunk elô... stb. stb. Sose felejtjük el, hogy minél nagyobb szabadságot kap tólünk a játékos, annál jobban élvezi majd a játékot - igaz, másik fele a dolognak, hogy ez sajnos nagyságrendekkel megnehezíti számunkra a játékprogram megtervezését és elkészítését...

Említettük: a kalandjáték bizonyos szempontból a valóságos világ szimulációjára törekszik. Rendkívül sokat árt ennek a szándéknak, hogy bizonyos konkrét célok elérésére irányuló vonalakba rendezôdik a cselekmény - azt a hamis illúziót keltve, mintha az egész világnak egy bizonyos célja és értelme lenne... Ha a játékos elérte a végsô célját, akkor egyszercsak mintha elvágta volna: nincs tovább, leállt, filmszakadás, dráma... Attól kezdve már nem is érdekelheti tovább az egész. Segíthet ezen valamennyire a sokféleség, az, hogy mindennek többféle megoldása van; így legalább van értelme mindig újratekinteni, és az újabb és újabb változatokat próbálni, fölfedezni az elrejtett, mellékes kis részleteket is. De a legjobb (és persze legnehezebben elképzelhetô) megoldás az volna, ha egyáltalán nem lenne sem eleje, sem pedig vége az eseményeknek. Ez esetben ténylegesen is föl lehetne használni a valóság jelenségeinek ábrázolására. Lehetne például egy programot írni, ami - valóságos tényekre és tapasztalatokra építve - megjelenítené egy erdô mindennapos életét. Az alapszintér sokezer, egymáshoz nagyon hasonló helyszínbôl állna, amely lassan és fokozatosan, de nem szûnô következetességgel folyton-folyvást változna: bizonyos ôsvényeket lassanként benônne a csalán, míg elôbb-utóbb teljesen eltûnnének, s eközben máshol új, friss csapások képzôdnének az állatok lábai nyomán - nem lehetne egy állandó térképet rajzolni a világunk fölépítésérôl, mert mire elkészülnék ezzel, addigra talán már teljesen megváltozna a helyiségek elrendezése; sôt, maguk a helyiségek is eltûnnének idôvel, hogy azután hasonlóak, de nem pontosan ugyanolyanok bukkanjanak fel valahol egy másik pozícióján. Éppígy cserélôdnének a bennüket megtöltô tárgyak és élôlények is: természetesen ehhez min. hetekig vagy hónapokig kellene futtatni a programot, de ennyi idô alatt az oszlopos fák is szépen kiöregednének és kidôlnének (ha szerencsés a fôhês, akkor éppen olyankor tartózkodik egy ilyen érdekes esemény helyszínén, amikor az megtörténik, s így néha-néha közvetlenül figyelemmel kísérheti azt), hogy aztán újabb hosszú ideig korhadjanak a földön, mialatt a friss nyiladékban fejlôdik a következô nemzedék. A keletkezett tisztásokat benônék a cserjék és a virágok, rajtuk madarak és rovarok csapataival, esô után elôbújnak a csigák és szeptemberben bôgnének a szarvasok... Természetesen az idô múlása is figyelemmel kísérhetô lenne - alaposan felgyorsítva, pl. egy másodperc alatt múlna el ott egy perc, egy perc alatt egy óra, következésképpen 24 perc alatt egy nap; a napszakok váltakozásai nyomán változnának a helyiségek leírásai és a bennük található élôlények összetétele; kb. fél nap leforgása alatt pedig már egy teljes hónap peregne le a szemeink elôtt - hat nap alatt telne el egy év... Ezalatt mind a négy évszak megtenné a magáét: téli álomba vonulnának a medvék, leesne a hó és a fák lombjai, februárban malacokat ellene a vaddisznó, majd ezután a következô tavasszal újra kezdôdne minden. Ebben a játékban mászkálva nem azon mesterkedne a kalandor, hogy valamilyen célhoz kerüljön mindegyre közelebb, hanem elsôsorban a nagyságával és a gazdagságával nyûgözné le ôt: bármekkora kirándulótakat tenne is benne,

sohasem fedezhetné föl egészen, és képtelen volna egyszer s mindenkorra emlékezetben tartani annak a seregnyi állat- és növényfajnak a nevét, amelyek életciklusa mind-mind parányi részét nyújtaná a végleges képnek - köztük akár olyan ritkaságok is, amivel jó, ha egy életben egyszer találkozik az ember, vagy a teljes év folyamán mindössze néhány óráig vagy napig található szabadon... Sohasem lehetne megunni, elvégre örökké más-más lenne benne minden, és a kissé részletesebb megismeréshez is rendkívül hosszú ideig kéne foglalkozni vele. Bár őszintén megvallva, a valóságos természetjárás azért valamivel izgalmasabb időtöltés ennél... Nem is ez volna benne az igazi kihívás - hanem egy ilyen programot elkészíteni!

Ez volna az egyik út, amelyen tovább lehetne lépni innen; a másik pedig a játékok szöveges voltát használná ki, és magának a Nyelvnek a szimulációjára törne. Történtek is már ez irányban különféle kisebb-nagyobb próbálkozások; de az elfuserált fordító- és beszélgetőprogramok - minden beléjük fektetett szorgalmas és kemény munka ellenére is - igen szálnalmas látványt nyújtanak... Nem csoda, ugyanis ez még az előbbinél is jóval hatalmasabb feladat volna, és nem tudom, hogy egyáltalán megvalósítható-e. Roppant érdekes adalékok ehhez a témához Noam Chomskynak, az amerikai nyelvésznek és filozófusnak magyarul csak nemrég megjelent Mondattani szerkezetek valamint Nyelv és elme című írásai - előbbiben az emberi beszédnek egy pontos matematikai modelljét igyekszik fölállítani, összetevőkkel és ún. nyelvi transzformációk segítségével; utóbbiban azt foglalja össze, milyen hatással volt és lehet a nyelvészet az emberi elme kutatására. Számtan és nyelvtan összekapcsolására mások is tettek már kísérletet, pl. a magyar Kiss Dénes is - de az övé jóval kevésbé módszeres és alapos, mint a Chomskyé, inkább amolyan ösztönösen ide-oda csapongó.

4. IRODALOM

- Chomsky, Noam. (1995). Mondattani szerkezetek - Nyelv és elme. Osiris-Századvég, Budapest.
- DaCosta, F. (1986). A kalandprogram írásának rejtelmek. Műszaki Könyvkiadó, Budapest.
- Kiss, László és Schmidt, Endre. (1988). 1001/2 játék. LSI Alkalmazástechnikai Tanácsadó Szolgálat, Budapest.
- Tolkien, J. R. R. (1992). A babó. Ciceró.

Közreadta : a játékgyáros, aki már megszenvetde a poklokat, a Bölcsek Könyvével és Robot a Marsról szöveges, rajzos kalandjátékaival.

JÁTÉKTERV ÚTMUTATÓ

Hogyan is írjunk kalandjátékot ???

A fiókomból előkerült valami, ami csak régi iromány,- szöveges kalandjáték lett volna - annak idején meg is próbálkoztam vele, de aztán abbamaradt, elfogyott a lelkesedésem, az időm és a türelmem . Igen, ma már csak emlék, 1986-ból „A JÁTÉK „ (egy játékterv, melyből soha nem készült kalandjáték)

Bevezetés:

Az A JÁTÉK egy szöveges kalandjáték. Akinek a '82-90-es években volt 8 bites számítógépe (ZX-Spectrum, C64, TRS-80, HT-1080Z, és TVC), az bizonyára emlékszik az ilyen stílusú játékokra. Én emlékszem-mi emlékszünk! Rengeteg kalandot éltünk át úgy, hogy a helyszíneket csupán leírásokból, képek nélkül, a fantáziánkra hagyatkozva képzeljük el, ami különös hangulatot kölcsönzött az ilyen játékoknak. Ezt a hangulatot szeretném ismét felidézni, hogy a fiatal számítógépesek megismerhessék ezt, a régi motorosok pedig nosztalgizhassanak, és újra nekivághassanak az igazi kalandoknak!

Játékszabály:

A játékszabály elég egyszerű. Lépj be egy világba, teljesítsd a küldetést és próbálj meg nem meghalni! :) Persze ennél azért többre lesz szükséged a kalandokhoz. A játék szöveges kalandjáték, tehát a helyszíneket annak leírásából ismerheted meg. A helyszínek között mozogni a következő parancsok kiadásával lehet: /észak, /északkelet, /kelet, /délekelet, /dél, /délnyugat, /nyugat, /északnyugat, /fel, /le, /ugrik, /mászik. Ezen kívül a program még ismeri a /felvesz, /letesz, /vizsgál, /használ, /nyom, /húz, /aktívfegyver, /támad, /kő, /papír, /olló parancsokat. Ezek részletes leírása később. Tehát a világokban ezeket a parancsokat kiadva kalandozhatsz, a kiadott parancsokra jött választ pedig a képernyőről, szöveges formában, leolvashatod. A helyszíneken akadályok nehezítik majd a továbbjutásodat, amiket különböző tárgyak segítségével győzhetsz le. Az is lehet, hogy egy másik (agresszív) kalandozó megtámad. Erre az esetre jó, ha van nálad fegyver! Utad során tehát négyféle dologgal találkozhatod szembe magadat: -akadály -tárgy -fegyver -egy másik kalandozó Az akadályok elég különbözőek lehetnek, kezdve a mocsártól, a sötétségen át, egészen egy őrig, bármi. Viselkedésük is sokféle lehet. Például a sötétség lámpa nélkül azonnal megöl, mert mondjuk elesel, és kitöröd a nyakad. A földön lévő sav nem öl meg azonnal, de saválló cipő nélkül folyamatosan szívja le az energiádat. Egy bezárt ajtó nem "bánt", de ha nincs kulcsod, akkor nem tudsz továbbmenni. Az akadályok működése elég egyszerű: ha nincs nálad egy bizonyos tárgy, akkor meggátol a továbbjutásban. Tehát mindenképpen tárgyakra lesz szükséged, amiket el kell cipelned az akadályig, hogy átjuthass rajta. Minden tárgynak van súlya. A tárgyat a hátizsákodba rakhatod, ami persze csak bizonyos súlyt képes elviselni, így néha nem árt a már feleslegessé vált tárgyat eldobni. Ha a megfelelő tárgy nálad van, az akadály nem lesz akadály többé számodra. Például, ha a kulcs nálad van, az ajtó tovább enged. (nincs szükség semmilyen parancs kiadására) Találhatsz olyan tárgyat is, amelyek semmilyen akadályhoz nem szükségesek, viszont van energiájuk, amik a te energiádat növelhetik. Például a gyógyszer, étel, ital. Ezeket felvéve, ha megsérültél növekszik az energiád. Érdemes tehát ilyen tárgyak után is kutatni! Ezek a tárgyak különösen harc után tehetnek jó szolgálatot. Ha harcolsz, akkor fegyverre is szükséged lesz! Különböző fegyvereknek, különböző ereje lehet, így célszerű minél jobb fegyvert beszerezni, és azt aktiválni. Sose felejtse el aktiválni a felvett legjobb fegyveredet! Hogyan tudsz felvenni, letenni tárgyat, megtámadni valakit, harcolni, és fegyvert aktívvá tenni? Természetesen parancsokkal, ezért hát nézzük meg a parancsokat most már részletesen!

Parancskészlet:

Parancsok kétféleképpen adhatók ki. Az egyik lehetőség, hogy beírod őket. Ekkor a parancsokat mindig / jellel kell kezdeni, és minden hozzájuk írt egyéb információt (tárgy neve, személy neve, akadály neve) szintén / jellel kell a parancsszó után írni. A másik lehetőség a képernyőn látható gombokat megnyomva kiadni az utasítást. A játékban a következő parancsokat adhatod ki:

1.) Mozgás parancsok (/észak, /északkelet, /kelet, /délekelet, /dél, /délnyugat, /nyugat, /északnyugat, /fel, /le, /ugrik, /mászik) Ezekkel a parancsokkal tudsz mozogni az egyes helyszínek között. Ezek az égtájak

nevei, kiegészítve a fel,le, ugrik, mászik irányokkal. Hogy milyen irányba mozoghatsz, az kiderül a helyszín leírásából. A parancsokat egyszerűen beírva és ENTER-t ütve, vagy a megfelelő ikonra rákattintva a parancs végrehajtódik. A vezérlőpulton található nyilak az egyes égtájakat jelölik, tehát:

Észak
Északkelet
Kelet
Délkelet
Dél
Délnyugat
Nyugat
Északnyugat

A nyilak mellett látható ikonsor első négy ikonja balról jobbra a következők:

Fel
Le
Ugrik
Mászik

2.) Felvesz és letesz parancs (/felvesz, /letesz) Működése: /felvesz/[a tárgy neve, amit fel akarsz venni] /letesz/[a tárgy neve, amit fel akarsz venni] Például: /felvesz/alma vagy /letesz/alma Vagy a

Felvesz
Letesz

ikonra, majd egy tárgy vagy fegyver nevére kattintva. Felvesz esetében a szobában található tárgyak listájából, letesz esetén a hátizsákban lévő tárgyak listájából kell választani. E két parancs segítségével fel tudsz venni és le tudsz tenni egy tárgyat. A felvett tárgy a hátizsákodba kerül. A felvett tárgy súlya hozzáadódik a hátizsákban lévő tárgyak súlyához. Ha a felvenni kívánt tárgy súlyával együtt a hátizsák nehezebb lenne, mint az előre meghatározott maximális hátizsáksúly (ami alap esetben 100), akkor a tárgyat nem fogod tudni felvenni.

3.) Vizsgál parancs (/vizsgál) Működése: /vizsgál/[a tárgy neve, amit meg akarsz vizsgálni]

Például: /vizsgál/ajtó Vagy a Vizsgál ikonra, majd egy objektum nevére kattintva a szoba objektumainak listájában. Ezzel a paranccsal bővebb információ kérhető a szobában lévő objektumokról.
(Tárgy, fegyver, akadály)

4.) Használ parancs (/használ) Működése: /használ/[mit]/[mivel] Például:

/használ/üres zseblámpa/elem Vagy a Használ ikonra, majd egy objektum nevére kattintva a szoba objektumainak listájában vagy a hátizsákban lévő tárgyak listájában, majd egy másik objektum nevére kattintva a szoba objektumainak listájában vagy a hátizsákban lévő tárgyak listájában. A parancs hatására két tárgy használatával egy új tárgy fog keletkezni (pl.: üres zseblámpa + elem = működő zseblámpa vagy kólaautomata + pénz = kóla).

5.) Nyom és húz parancs (/nyom, /húz) Működése: /nyom/[a tárgy neve, amit meg akarsz nyomni] /húz/[a tárgy neve, amit meg akarsz húzni] Például: /nyom/gomb vagy /húz/kar Vagy a

Nyom
Húz

ikonra, majd egy objektum nevére kattintva a szoba objektumainak listájában vagy a hátizsákban lévő tárgyak listájában. Hatására a helyszín fog megváltozni. Új akadály keletkezik, vagy éppen eltűnik. Esetleg valamilyen tárgy jelenik meg.

6.) Aktív fegyver parancs (/aktív fegyver) Működése: /aktív fegyver/[a fegyver neve, amit aktiválni akarsz] Például: /aktív fegyver/puska Vagy az Aktív fegyver ikonra, majd egy fegyver nevére kattintva a hátizsákban lévő tárgyak listájában.

Fontos, hogy a kiválasztott fegyvernek a hátizsákban kell lennie! Ha egy fegyvert aktívvá teszed, akkor amikor harcba keveredsz, és nyersz, az ellenfeleden olyan mértékű sérülést tudsz okozni, mint amilyen az aktív fegyvered ereje. A fegyver erejét a hátizsák listában lévő, a fegyver megnevezés után látható második szám jelzi. Az első szám a tárgy energiáját jelenti. Erre akkor van szükség, ha téged sebesítenek meg. Ekkor a tárgyaink energiája hozzáadódik az energiádhoz, ezáltal a sérülés kisebb lesz, esetleg meg se fogsz sérülni. (Például: puska (0 - 30) azt jelenti, hogy a puska, egy sikeres forduló után 30 pontot fog levonni az

ellenfeled energiájából; gyógyszer (40 - 0) azt jelenti, hogy a gyógyszer 40 ponttal tudja növelni az energiádat egy esetleges sérülés után)

7.) Harci parancsok (/támad, /kő, /papír, /olló) Működése: /támad/[kit] A kő,papír, olló önmagában működik, semmiféle plusz paramétert nem várnak. Például:

/támad/jános Vagy a

Támad

Kő

Papír

Olló

ikonra kattintva. Támad esetén még ki kell választani egy nevet a helyszínen lévő játékosok közül, és rákattintani. Ha megtámadtál valakit, akkor az ismert kő, papír, olló játékot játszva dönthetitek el, hogy az adott fordulót ki nyerte. Ha valaki nem ismerné a szabályait: a kő kicsorbítja az ollót (tehát, ha az első játékos a követ választja, a második az ollót, akkor az első játékos nyer), az olló elvágja a papírt (tehát, ha az első játékos az ollót választja, a második a papírt, akkor az első játékos nyer), a papír becsomagolja a követ (tehát, ha az első játékos a papírt választja, a második a követ, akkor az első játékos nyer). A vesztes játékos energiájából, a nyertes játékos aktív fegyverének ereje levonódik. A csata addig folytatódik, amíg valamelyik játékos meg nem hal, vagy el nem menekül. A menekülésnek viszont ára van! Ha valaki elmenekül, akkor az adott fordulót elvesztette, tehát az ellenfele fegyverének az ereje le lesz vonva az energiájából.

Most már ismerjük a játék parancsait, lássuk, hogy néz ki a játék:

A képernyő alján legyen a vezérlőpult. A vezérlőpulton belül, a felső részben, vörös háttérben látható egy szövegbeviteli mező. Ide lehet a parancsokat beírni, és ennek segítségével tudsz beszélgetni az adott helyszínen lévő többi játékosal. Baloldalon és alul, zöld háttérben láthatóak a vezérlőikonok, a már említett sorrendben. Jobb oldalon, kék háttérben látható az energiádat, a hátizságod jelenlegi súlyát, és az éppen kiválasztott, aktív fegyveredet. A vezérlőpult fölött öt doboz van. A jobb oldali dobozban jelennek meg azoknak a játékosoknak a nevei, akik veled egy helyszínen tartózkodnak, tehát akikkel beszélgetni, vagy éppen harcolni tudsz. A közepén, felül lévő dobozban jelenik meg a helyszín leírása, és a különböző rendszerüzenetek. Például, ha megvizsgálsz egy tárgyat, akkor a tárgy leírása itt fog megjelenni. Középen, alul van a chat doboz. Itt a beszélgetések olvashatóak, a te és a helyszínen tartózkodó többi játékos üzenetei. Jobb oldalon fölül jelennek meg a szobában található tárgyak, fegyverek, akadályok. Ha például fel szeretnél venni egy tárgyat, kattints a felvesz ikonra, majd utána ebben a dobozban az adott tárgy nevére. Alatta, jobb oldalt van a hátizságodban található tárgyak listája. A tárgyak vagy fegyverek neve mellett zárójelben láthatsz két számot. Az első azt mutatja, hogy az adott tárgy (vagy fegyver) mennyi energiát tartalmaz, a második szám pedig azt, (fegyver esetén) hogy mekkora a fegyver ereje. Például az "alma (5-0)" azt jelenti, hogy az alma 5 pontot tud az energiádhhoz adni, a "kard (0-15)" pedig azt, hogy ha a kard az aktív fegyvered, akkor harc közben 15 pontot fog levonni az ellenfél energiájából egy nyertes forduló esetén. Ha például el szeretnél dobni egy tárgyat, akkor kattints a letesz ikonra, majd ebben a dobozban annak az adott tárgy nevére.

Általánosan ennyi a játék. A világok leírásáról, a küldetésről, esetleg a speciális tudnivalókról minden világ első helyszínén olvashatsz.

Bevezetés:

Írj te is egy világot!

Hogyan kell (lehet) kalandjátékot írni ??

1.) Háttér kitalálása. Mielőtt belefognánk egy új világ megírásába, lássuk, hogy általánosan hogyan is kell felépíteni egy kalandjátékot. Első lépésként találj ki egy hátteret. Egy helyet, amibe belehelyezed a történetet. Ez a hely bármi lehet, például egy sötét barlangrendszer, ahol szörnyekkel találkozol majd a kalandozó, vagy egy katonai bázis, ahonnan titkos iratokat kell kicsempészni, vagy egy ismeretlen bolygó, ahonnan az űrhajójával lezuhant játékosnak vissza kell térnie. Tehát bármi lehet, a lényeg, hogy megfelelően tágas legyen, vagyis legyen elegendő hely a mozgásra. Az A JÁTÉK esetében a sok helyszín különösen fontos, hiszen nem szabad elfelejteni, hogy a világhálóról egyszerre sok ember léphet be a világba, és nem jó, ha ez a világ a kevés helyszín miatt "túlzsúfolt".

2.) Küldetés kitalálása. Ha már megtaláltad a legjobb helyet, akkor kell egy küldetés a játékosnak. Például a barlangrendszer közepén meg kell találnia a világ legnagyobb gyémántját, a katonai bázisból el kell hozni a titkos iratokat, az idegen bolygón kell találni egy használható űrhajót, és azzal visszatérni a földre.

3.) Összefüggések kitalálása. A játék lényege azonban az akadályok legyőzése, amikkel a játékos a kalandozása során találkozik. Az akadályok sokfélék lehetnek. Például szörnyek, amiket meg kell ölni, ajtók, amiket különböző kulcsokkal ki kell nyitni, de akadály egy folyó, amin csónak nélkül a játékos nem tud átkelni, vagy akadály a sötétség, amin a játékos lámpa nélkül nem tud átjutni. Akkor lesz izgalmas a játék, ha ezek az akadályok összekapcsolódnak, és egyiken, csak a másik legyőzésével tud a játékos átmenni. Például legyen a feladatunk (küldetés), hogy be kell jutnunk egy házba, aminek az ajtaja zárva van. Tehát valószínűleg kell egy kulcs. A pályán a következő helyszínek vannak:

a ház bejárata, egy kert, magas fűvel és gazzal, egy fészer, egy kút, és a kert vége, ahol szárítókötélen ruhák száradnak. Hol lehet a kulcs? Először elmegyünk a kertbe. Megvizsgáljuk a kertet, amire azt a választ kapjuk, hogy nagyon magas a fű, le kéne már vágni. Továbbmegyünk a fészertől. Ezt is megvizsgáljuk, és látjuk, hogy egy rozsdás lakattal van bezárva, ami olyan rozsdás, hogy ezt kulcs már nem nyitja. Továbbmegyünk a kúthoz, amit szintén megvizsgálunk. Mély kút, elég romos állapotban. Amikor le akarunk mászni, kiderül, hogy nem tudunk, mert már a kötélen sincs a kúton, amin leereszkeszhetnénk. Menjünk tovább a kert végébe.

Itt ruhák száradnak, szárítókötélen. Megvizsgáljuk a szárítókötélet, ami jó erős. Hát felvesszük, és már rohanunk is vissza a kúthoz. A kötélen segítségével leereszkeszünk a kút aljára, ahol találunk egy kalapácsot. A kalapáccsal visszamegyünk a fészertől, leütjük a lakatot az ajtóról, és bent találunk egy kaszát. Átmelegyünk a kertbe, lekaszáljuk a fűvet, és a fű között megtaláljuk a ház kulcsát. Visszamegyünk az ajtóhoz, és bemegyünk a házba. Ez egy nagyon rövid kaland volt, viszont jó példa arra, hogyan ágyazódnak egymásba az egyes akadályok. Nem lehet továbbjutni, csak lépésről lépésre. Ennek a kalandnak az összefüggései a következők voltak: szárítókötél -> kút -> kalapács -> fészer -> kasza -> fű -> kulcs -> ajtó. Ilyen összefüggésekből lehet felépíteni a játékot.

A kaland akkor izgalmas, ha mondjuk, az ajtó mögött újabb összefüggéseket kell majd megfejtenünk úgy, hogy annak egy-egy láncszemét már itt a kertben megtaláljuk. Tehát az összefüggések a végtelenségig bonyolíthatóak. Minél bonyolultabbak, annál több fejtörést fog okozni a játékosoknak, és annál izgalmasabb lesz a kaland.

A program sajátosságai:

Lássuk hát, hogyan is működik az A JÁTÉK, és hogyan lehet vele új világot alkotni. A program legyen átlagos kalandjáték, legyen itt egy időben több játékos is, ugyanabban a világban, sőt, ugyanazon a helyszínen. Így több furcsasága is lehet a játéknak. Ezért például az, hogy ebben a játékban a hátizsákból kidobott tárgy nem jelenik meg az adott helyszínen, hanem eltűnik.

Ez egy kicsit furcsa lehet, de azért volt rá szükség, mert ha az eldobott tárgyak megjelenének azon a helyszínen, ahol letettük őket, akkor hamarosan (mivel többen is játszhatnak ugyanabban a világban) az egész pálya tele lenne tárgyakkal. A másik ilyen furcsaság, hogy a felvett tárgy, miután felvettük, a földön marad. Bekerül a hátizsákba, de nem tűnik el a pályáról. Ezt is a többjátékos üzemmód miatt kell megengedni, azért, hogy mindenkinek legyen esélye végigvinni a küldetést.

Hiszen gondoljuk végig, ha a földön fekvő kulcs, miután felvettük eltűnne a földről, akkor már senki más nem tudna továbbmenni, hiszen senki más nem jutna hozzá a kulcshoz. A harmadik dolog, ami furcsa lehet a játékban az, az akadályok működése. Az ajtós példánál maradva, ha a kulcs nálunk van, ki tudjuk nyitni a bezárt ajtót, és át tudunk menni az akadályon. Azonban, ha a kulcsot valahol letesszük, és ismét visszakerülünk az iménti ajtó elé, az ismételten nem fog minket átengedni, hiába nyitottuk már egyszer ki. Egyszerűen tehát minden akadály csak addig van "nyitva", míg a leküzdéséhez szükséges tárgy a hátizsákban van. Könnyen belátható ennek is az oka. Ha minden akadály eltűnne, miután valaki legyőzte, akkor a sokadik játékos, aki belép a játékba, már nem találna leküzdendő akadályt, és minden gondolkodás nélkül átsétálhatna a már teljesen semlegesített pályán. Ezekre a dolgokra tehát oda kell figyelni.

Egy világ megtervezése:

Mielőtt leülnénk a gép elé, papíron tervezzük meg a játékot. Kitaláltuk már a háttérrel, azt, hogy hol fog játszódni a kaland, tudjuk a küldetést, azt, hogy mi a játék célja, mit kell megszerezni, hova kell eljutni, és végül az összefüggéseket is kitaláltuk, azt, hogy hogyan és milyen tárgyak megszerzésével, milyen akadályok legyőzésével lehet teljesíteni a küldetést. Elsőként tehát össze kell írni az összefüggések alapján az objektumokat (tárgy, fegyver, akadály). Írjuk föl egy lapra az objektumaink nevét, és mindegyiknek adjunk egy számot. A pályaszerkesztés során az objektumainkra majd ez a szám alapján tudunk hivatkozni. A második feladat a tárgyak, akadályok és fegyverek elhelyezése a pálya helyszínein. Ehhez három lapra lesz szükség. Az első lap sarkába rajzoljunk be egy iránytűt. Felfele legyen észak, lefele dél, jobbra kelet,

balra nyugat. Ez lesz maga a térkép. A térképen majd minden helyszínt be fogunk számozni. A második lapra fogjuk írni a helyszínek leírását, a számozás alapján azonosítva, hogy a térkép melyik helyszínéhez, melyik leírás tartozik. A harmadik lapon egy táblázat lesz. Ebben a táblázatban fogjuk feltüntetni, hogy melyik szobában milyen objektumok (tárgy, akadály, fegyver) vannak, melyikből hány darab van, és mozdíthatóak-e, vagy nem. Ha ezekkel is megvagyunk jön majd a különböző cselekvések feljegyzése. A cselekvések, a használ, nyom-húz, vizsgál parancsok. Olyan táblázatokat kell majd csinálnunk, ahol feljegyezzük, hogy hányas számú tárgyra kiadva az adott parancsot, mi fog történni. A szerkesztés könnyebb megértése érdekében a folyamatot egy egyszerű pár helyszínes világon keresztül fogom bemutatni. Egy üzemképes játékhoz persze ennél majd jóval több helyszínes és bonyolultabb világot kell írnotok, de példának, és a könnyű érthetőség miatt lesz a világ ilyen kicsi. Kezdjük is el!

Cím: Rossz nap.

Háttér: Jelen kor, egy kertés ház, és környéke.

Küldetés: Hazatérve, az ajtód előtt veszed észre, hogy elvesztetted a kulcsodat. Kulcs nélkül nem jutsz be a házba, hiszen épp tegnapelőtt szereltettél fel betörésbiztos ajtót valami céggel, akik nagyon reklámozták magukat, és a bemutatójuk, hogy még dinamittal sem lehet berobbantani az ajtót, nagyon meggyőzőtt. Valahol van egy pótkulcs, de már nem emlékszel rá hol. Meg kell találni, hogy bejuthass a lakásodba.

Összefüggések: ajtó <- kulcs <- (húz) <- fiók <- fészer (az ajtó nem nyílik, de "fent magasan van egy ablak") <- létra <- (/használ/festő/sör) <- festő (Hú de meleg van! Bármit odaadnék egy üveg sörért!) <- sör <- (/használ/automata/pénz) <- automata <- pénz <- szarkafészek <- fa (a kert közepén egy magas fa áll)

Túl vagyok hát az első három legfontosabb feladaton. Kitaláltam egy háttérrel, egy küldetést, és az összefüggéseket is felírtam, ami alapján majd végig lehet vinni a küldetést. Ahogy látható hátulról kezdtem építeni az összefüggéseket. Azzal a tárggyal kezdtem a sort, amit majd legutoljára talál meg a játékos.

Zárójelbe tettem pár emlékeztetőt. Például a fiók és kulcs közé a húz szót, ami azt jelenti, hogy amikor a játékos belép a fészerbe, ott talál egy szekrényt, vagy asztalt, aminek lesz egy fiókja. Ha itt kiadja a /húz/fiók parancsot, akkor megjelenik a kulcs. Addig természetesen nem fogja látni a kulcsot a fészerben.

Az összefüggések listájában jól megfigyelhető a használ parancs "furcsasága". Az, hogy /használ/automata/pénz mindenki számára érthető. Használni akarom a pénzt az automatával, azaz bedobom a pénzt az automatába. De ugyanezt a parancsot kell akkor is használni, amikor a festőtől akarom megszerezni a létráját, és ezért odaadom neki a sört. Ezt is a /használ/festő/sör paranccsal oldom meg. Itt nyelvi szempontból értelmesebb lenne egy /ad/festő/sör parancs, de logikailag az ad és a használ is ugyanúgy működik, és a használ parancsot kiadva is mindenki érti, hogy miről van szó. A sört akarom használni. Úgy használom, hogy odaadom a festőnek. Az összefüggések kitalálásakor vigyázzunk arra, hogy a használ, nyom, húz parancsok, új tárgyakat tudnak létrehozni.

Esetünkben a fiókot kihúzva kulcs fog "keletkezni", a pénzt használva az automatával pedig egy üveg sör fog megjelenni. Mindhárom parancs tud új tárgyat létrehozni. Az új tárgyat vagy az adott helyszínre tesszük, vagy a játékos hátizsákjába. Figyelni kell arra, hogy a továbbjutáshoz elengedhetetlenül szükséges tárgyak ne a helyszínen jelenjenek meg, hiszen így bárki felveheti. Ne feledjük, több játékos is lehet egy időben, ugyanazon a helyszínen! És elég bosszantó, ha én rájövök valamilyen rejtélyre, megoldom, és végül az ennek eredményeként megjelenő tárgyat nem tudom felvenni, mert valaki, aki szintén velem egy helyszínen tartózkodik, és gyorsabban beírja a parancsot (vagy gyorsabb Internet-kapcsolata van), felveszi előlem azt, amiért én dolgoztam meg.

Térkép (helyszínek vagy szobák)

Most már tudom, hol játszódik a játék, és milyen tárgyak lesznek elhelyezve rajta. A következő feladat a helyszínek megtervezése. Fogok egy lapot, ahogy már az előbb leírtam, felrajzolok egy iránytűt az egyik sarkába. A helyszíneket körökkel jelölöm, amiket majd vonalakkal kötök össze. A vonalak jelölik, hogy milyen irányba lehet menni. Lerajzolva ez rögtön érthető lesz. Minden kört, azaz helyszínt megszámozok. Az egyes sorszámúnak mindig a kezdő helyszínnek kell lennie, mivel minden új belépő játékos, az egyes sorszámú szobából indul. Ez azt jelenti, hogy a kerettörténetet (háttérrel, küldetést) az egyes sorszámú helyszín szövegének tartalmaznia kell. A legjobb megoldás talán az, ha az egyes helyszín egy olyan speciális helyszín, ami csak a kerettörténetet tartalmazza, és az összes irány (észak, dél, kelet, stb..) a kettes sorszámú helyszínre mutat, ahonnan viszont már nem lehet visszamenni az egyes sorszámú helyszínre. Tehát az egyes sorszámú helyszín szövegéből a játékos megtudja mi a feladata, utána kilép innét, és a tényleges játék a kettes sorszámú helyszínről indul. Ugyanilyen speciális az utolsó helyszín is. Ennek a

szövegébe kell kerülni a befejezésnek. Ide jön a gratuláció, hogy a játékos sikeresen teljesítette a küldetését. Viszont jó, ha innét már nincs visszaút. A kalandjátékok egy kedvelt fogása az útvesztő. Az útvesztők kitágítják a világot.

Működése nagyon egyszerű. Általában három-négy helyszínből óriási útvesztőt lehet csinálni. Lényegük, hogy egy, esetleg két kijáratuk van, viszont mindegyik helyszínből, minden irányba lehet menni, és mindegyiknek ugyanaz a leírása. Például: egy sötét erdőben bolyongsz, vagy egy végeleáthatatlan bűzamezőn sétálsz. Viszont nem logikusak az égtájak szerinti mozgások. Például van három helyszínem, a 10-es, 11-es és 12-es. A kijárat-bejárat legyen a 10-es helyszín keleti iránya. Északkelti irányba mozogva a 11-es helyszínre jut a játékos. De a 11-esből délnyugati irányba mozogva, nem jut vissza a 10-esbe, hanem mondjuk ez az irány visszamatat a 11-es helyszínre. A 11-esből nyugatra mozogva a 12-esbe jutunk, de a 12-esből keletre, a 10-esbe. A 10-esből nyugatra mozogva visszajutunk a 10-es helyszínre. És így tovább. Mivel mindhárom helyszín szövege ugyanaz, a játékos nem fogja tudni, hogy hol is van tulajdonképpen. A térképpel együtt két táblázatot is készítek. Egyikbe írom a helyszínek leírását, másikba, hogy melyik helyszínen milyen objektumok vannak.

Helyszínek:

- 1.) Hazatérve, az ajtód előtt veszed észre, hogy elvesztetted a kulcsodat. Kulcs nélkül nem jutsz be a házba, hiszen épp tegnapelőtt szereltettél fel betörésbiztos ajtót valami céggel, akik nagyon reklámozták magukat, és a bemutatójuk, hogy még dinamittal sem lehet berobbantani az ajtót, nagyon meggyőzőtt. Valahol van egy pótkulcs, de már nem emlékszel rá hol. Meg kell találni, hogy bejuthass a lakásodba.
- 2.) A lakásod bejárata előtt állsz. Délre tőled az ajtó, északra az utca van. Kelet-nyugati irányba egy kövekből kirakott járda halad a házad előtt.
Délnyugatra egy kis ösvény fut.
- 3.) 4.) 5.) Az utcákon kóborolsz.
- 6.) A kerted közepén állsz egy magas fa mellett. Egy fészker van délnyugatra, a kert végében. Délre egy kis ösvény vezet a virágoskertedbe, keletre egy kövezett járda visz a lakásod bejáratához.
- 7.) A fa tetején vagy. Az ágak között egy szarka fészkére bukkantál.
- 8.) A házad keleti végében vagy. A fal félig átfestve.
- 9.) Az utcákon kóborolsz.
- 10.) A kert végében vagy, egy fészker mellett. Az ajtó beszorult, nem nyílik. Ösvények vezetnek északkeleti, keleti és déli irányba.
- 11.) A fészkerben vagy.
- 12.) A virágoskertedben vagy. Nyugatra, északra és északkeletre vezetnek ösvények.
- 13.) A kert déli végében vagy. Északra egy ösvény vezet a fészkerhez.
- 14.) Végre sikerült kinyitni az ajtót. Fáradtan beleroskadsz a kedvenc karosszékedbe, bekapcsolod a tévét, és lassan elalszol. Gratulálok! Sikeresen teljesítetted a küldetésedet! Jó pihenést! :)

Objektumok és elhelyezkedésük:

- 1.) pénz - 7
- 2.) automata - 9
- 3.) sör - 0
- 4.) festő - 8
- 5.) létra - 0
- 6.) ablak - 10
- 7.) fiók - 11
- 8.) kulcs - 0
- 9.) húsevő növény - 12
- 10.) magas szárú csizma - 13
- 11.) kés - 2
- 12.) fejsze - 6

Még egy olyan tárgy, amivel harc után az energiánkat növelhetjük meg.

- 13.) orvosság - 10

És egy olyan tárgy, ami csak azért van, hogy magyarázza egy másik tárgy jelenlétét:

- 14.) íróasztal (fiókkal!!!) - 11

Azok a tárgyak, amik azért kellene, mert a helyszín leírása utal a létezésükre:

- 15.) fa - 6
- 16.) fészker - 10
- 17.) fészkek - 7

És végül a kijáratot lezáró akadály:

- 18.) ajtó - 2

Az objektumok megnevezése után kötőjellel elválasztva annak a helyszínek a számát kell beírni, ahol az adott objektum található. A 0 azt jelenti, hogy ez az objektum egyenlőre sehol sincs. Például a sör majd csak akkor fog megjelenni, ha a játékos bedobja a pénzt az automatába.

Már csak az egyes objektumokhoz tartozó vizsgálatok vannak hátra:

- 1.) pénz - Ez bizony pénz.
- 2.) automata - Pénzért cserébe sört ad!
- 3.) sör - Ez egy üveg sör
- 4.) festő - Hú, de meleg van! Le kellett jönnöm a létráról, úgy megszedültem. Bármit odaadnék egy kis italért!
- 5.) létra - Jó magas létra
- 6.) ablak - Nagyon magasan van
- 7.) fiók - Az egy fiók.
- 8.) kulcs - A bejárati ajtó kulcsa
- 9.) húsevő növény - A kert tele van húsevő növényekkel. Megpróbálnak a bokádba marni!
- 10.) magas szárú csizma - Védi az ember egész lábszárát
- 11.) kés - Ez egy jó éles kés
- 12.) fejsze - Ez aztán a fegyver!
- 13.) orvosság - Segít, ha megsérülsz
- 14.) íróasztal - Van egy fiókja!
- 15.) fa - Ez egy jól megmászható fa!
- 16.) fészker - Ennek az ajtaja soha nem fog kinyílni. Az ajtó fölött, magasan egy nyitott ablak van.
- 17.) fészkek - Vajon mi van egy szarka fészkében?
- 18.) ajtó - Az ajtó be van zárva.

Ha a játékos beírja a "vizsgál" parancsot akkor ezek az üzenet fognak megjelenni. A "vizsgál" parancs segíthet rávezetni a játékost a megoldásra.

Térkép elkészítése

A térkép táblázatban minden egyes sor egy új helyszín. Az első oszlop mutatja a helyszín sorszámát, ami automatikusan növekszik egyesével. Figyeljünk oda rá, hogy a papíron megtervezett világ helyszíneinek sorszámait követve vigyük fel a helyszíneket. A második oszlopba a helyszín leírása kerül. Most tehát előveszem azt a lapot, amire felírtam a helyszínek leírását, és kitöltöm az egyes helyszín szövegét. Ez a következő volt: "Hazatérve, az ajtód előtt veszed észre, hogy elvesztetted a kulcsodat. Kulcs nélkül nem jutsz be a házba, hiszen épp tegnapelőtt szerelttél fel betörésbiztos ajtót valami céggel, akik nagyon reklámozták magukat, és a bemutatójuk, hogy még dinamittal sem lehet berobbantani az ajtót, nagyon meggyőzőtt. Valahol van egy pótkulcs, de már nem emlékszel rá hol. Meg kell találni, hogy bejuthass a lakásodba.". Ezzel meg vagyok. A további oszlopok az Észak, Északkelet, kelet, stb. irányokba lévő helyszínek sorszámait kerülnék. Ha az adott irányba nincs szoba, akkor oda 0-át kell írni. Hogy ezt megfelelően ki tudjam tölteni, elő kell vennem azt a lapot, amire a térképet rajzoltam. Az első szoba minden irányához azt írtam, hogy a 2-es szobába vezet, tehát az összes mezőt 2-essel fogom feltölteni. Amikor elkészültem megnyomom a sor végén látható "Rendben" feliratú gombot, és a sor fel is lesz véve a táblázatba. Most jön a kettes helyszín. A 2-es helyszín leírása: "A lakásod bejárata előtt állsz. Délre tőled az ajtó, északra az utca van. Kelet-nyugati irányba egy kövekből kirakott járda halad a házad előtt.

Délnyugatra egy kis ösvény fut.". A további oszlopok tartalma (ezek észak, északkelet, kelet, délkelet, dél, délnyugat, nyugat, északnyugat, fel, le, ugrik, mászik): 3, 0, 8, 0, 14, 12, 6, 0, 0, 0, 0, 0. Vagyis, ha a térképre nézünk, láthatjuk, hogy a 2-es helyszínről északi irányba haladva a 3-asra jutunk, északkeleti irányba nem lehet menni. Keletre a 8-as helyszín van, délkeletre nem lehet menni, délre a 14-esbe jutunk, délnyugat felé a 12-es van, nyugatra a 6-os, északnyugatra pedig nem lehet menni. A fel, le, ugrik, mászik irányok mindegyike 0, vagyis a 2-es helyszínről nem lehet ezekbe az irányokba haladni. Az oszlopok is ennek megfelelően vannak kitöltve. Érdekes lehet még megnézni a 3-as helyszín kitöltését is. Emlékezzünk vissza, ez az útvesztő egyik helyszíne! Itt vannak olyan irányok, amik visszavezetnek a 3-as helyszínre, hogy a játékost megzavarják. Tehát, sorszám: 3, leírás: "Az utcákon kóborolsz.". Az irányok pedig: 4, 5, 4, 3, 2, 3, 9, 3, 0, 0, 0, 0. A fel, le, ugrik, mászik irányokba itt sem lehet haladni, de minden egyéb irányba igen. Ha

megfigyeljük, a délkeleti irányhoz a 3-as lett rendelve, azaz a 3-as helyszínről délkeletre haladva ismét a 3-as helyszínen találja magát a játékos. A többi helyszínt is hasonlóképpen kell kitölteni.

Szavak összeírása (parancsok, objektumok)

Következő lépés a világ "szótárának" összeállítása. Össze kell írni az összes parancs és az összes objektumom nevét, hogy összeálljon a játékom szókincse.

A táblázatban öt oszlop legyen. Az első egy sorszám. A második a "Megnevezés" oszlop. Ide kerül a parancs, vagy a tárgy, fegyver, akadály neve. A következő a "Parancs" oszlop. Ide vagy 1-es vagy 0-s szám kerülhet. Az 1-es azt jelenti, hogy az adott sor egy parancs, a 0-s azt, hogy objektum. Mivel a parancsok fel vannak töltve, nekem a továbbiakban csak a 0-s beállításra lesz szükségem. A következő a "Jelentés" oszlop. Ide egy számot kell beírni. Ez a szám parancs esetén a parancshoz tartozó kezelő kódja (erről később), objektum esetén ez lesz a tárgy, fegyver vagy akadály azonosító kódja. Nagyon fontos, hogy objektum esetén mind a megnevezésnek, mind a jelentés kódnak egyedinek kell lennie! Nem lehet két azonos megnevezés, vagy két azonos jelentés kód. (Parancs esetén más a helyzet, hiszen ekkor a jelentés valamelyik előre definiált kezelőt azonosítja. A megnevezésnek azonban parancs esetén is egyedinek kell lennie.) Ezzel a kóddal kell majd hivatkoznom az objektumaimra a további táblázatokban. A következő a "Parancskód". Ennek csak parancs esetén van jelentősége (később). Objektum esetén mindig 0 legyen! Most előveszem azt a lapot, amire felírtam az objektumaimat. Emlékezzünk vissza, ez az "Objektumok és elhelyezkedésük" lista. Elkezdem tehát felvinni az első objektumomat Az oszlopok tartalma tehát: Sorszám - 29, Megnevezés- Pénz, Parancs - 0, Jelentés - 1, Parancskód - 0. A következő: Sorszám - 30, Megnevezés - Automata, Parancs - 0, Jelentés -2, Parancskód - 0. A következő: Sorszám - 31, Megnevezés - Sör, Parancs - 0, Jelentés -3, Parancskód - 0. A következő: Sorszám - 32, Megnevezés - Festő, Parancs - 0, Jelentés -4, Parancskód - 0. És így tovább. Néhány szó még a parancsokról. A parancsokat kezelők értelmezik. Ezek a következők: 1-mozgás, 2-felvesz, 3-letesz, 4-vizsgál, 5-használ, 6-nyom/húz, 7-aktív fegyver, 8-támad, 9-harc, 10-leltár, 11-ismét, 12-energia, 13-aktív, 14-világból. Ha például kedvezni szeretnék azoknak, akik kizárólag csak gépelik a parancsokat, akkor felvehetek rövidítéseket. Például az irányokat kibővítem, és felveszek é, ék, k, dk, d, dny, ny, ény, f, l, u, m parancsokat is. Ennek menete: Megnevezés - é, Parancs - 1, Jelentés - 1, Parancskód - 1, Megnevezés - ék, Parancs - 1, Jelentés - 1, Parancskód - 2, Megnevezés - k, Parancs - 1, Jelentés - 1, Parancskód - 3 ... Megnevezés - u, Parancs - 1, Jelentés - 1, Parancskód - 11, Megnevezés - m, Parancs - 1, Jelentés - 1, Parancskód - 12. Így kibővítem a szavak táblázatát, a játékban a mozgáshoz elég lesz beírni a /é, vagy /m parancsokat a /észak és /mászik helyett.

Objektumok összeírása:

Következő lépés az objektumok tulajdonságainak a meghatározása. E táblázatban 3 oszlop legyen. Az első az "Objektumszám". Ide kerül a beállítani kívánt objektum száma. Második a "Típus". Ez 0, 1 illetve 2 lehet. A 0 azt jelenti, hogy az objektum tárgy. Az objektumok legnagyobb része tárgy lesz.

Tárgy minden, ami nem fegyver és nem akadály. A tárgy lehet felvehető vagy mozdíthatatlan. Semlegesíthet egy akadályt, vagy egy másik tárgyat használva, új tárggyá alakulhat, vagy új tárgyat hozhat létre. Az 1 azt jelenti, hogy az objektum fegyver. A fegyver csak mozdítható lehet! A 2 azt jelenti, hogy az objektum akadály. Az akadály nem lehet felvehető, minden esetben mozdíthatatlannak kell lennie! Az akadályok fogják nehezíteni a játékos útját.

Az utolsó a "Mozdítható" oszlop. Ha 0, akkor az adott objektum nem mozdítható, ha 1, akkor mozdítható. Az Objektumszám mezőbe, a Szavak táblázat Jelentés mezőjébe írt kód kerül. Tehát: Objektumszám - 1, Típus - 0, Mozdítható - 1. Ezzel meghatároztam, hogy az 1-es jelentéskódú szavam, ami a "pénz", tárgy típusú, felvehető objektum. A következő: Objektumszám - 2, Típus - 0, Mozdítható - 0. Ezzel meghatároztam, hogy a 2-es jelentéskódú szavam, ami az "automata", tárgy típusú, nem felvehető objektum. Most egy akadály: Objektumszám - 6, Típus - 2, Mozdítható - 0. Ezzel meghatároztam, hogy a 6-os jelentéskódú szavam, ami az "ablak", akadály típusú, nem felvehető objektum. Végül egy fegyver: Objektumszám - 11, Típus - 3, Mozdítható - 1. Ezzel meghatároztam, hogy a 11-es jelentéskódú szavam, ami a "kés", fegyver típusú, felvehető objektum. Meghatároztam tehát, hogy melyik objektum milyen típusú. Most típus szerint meg kell adnom a speciális jellemzőket.

Menjünk sorban.

Első az objektumszám. Annak az objektumnak (ebben az esetben már csak tárgy típusú lehet) a száma, aminek a speciális tulajdonságait be akarom állítani.

Második a súly. Ez a tárgy súlya. Bármilyen szám lehet, de azért ne essünk túlzásba. A világ létrehozásánál lehet beállítani, hogy mi legyen az a maximális súly, amit egy játékos a hátizsákjában elbír. Most ezt ennek

az arányában kell meghatározni. Az én világomnak 100 volt a maximális hátizsákban vihető súly értéke, tehát nem fogok 120-as értéket megadni egy tárgynak, mert akkor azt nem lehet felvenni. Arra is figyelni kell, hogy például a pénz ne legyen nehezebb, mint mondjuk a létra. Az utolsó oszlop a tárgyenergia. Ennek azoknál a tárgyaknál van jelentősége, amik gyógyító erejűek. Ha a játékos megsérül, és felvesz egy olyan tárgyat, aminek van tárgyenergiája, akkor ez az energia hozzáadódik az energiájához. A maximális 100-as energiánál azonban soha nem lesz több a játékos energiája. Tehát, ha a játékos felvesz egy 10-es tárgy-energiájú tárgyat, mondjuk kenyeret, és az energiája 70, akkor a kenyér tárgy-energiája 0 lesz, a játékos energiája viszont 80. De ha a játékos energiája 95, és felveszi a 10-es tárgy-energiájú kenyeret, akkor a kenyér energiája 5 lesz, a játékosé 100. Olyan tárgyakhoz, mint például a létra, nem érdemes tárgy-energiát rendelni. Ez esetben ez az érték 0 legyen. Nézzünk két példát: Objektumszám - 1, Súly - 1, Tárgyenergia -0. (Tehát a pénz súlya 5, energiája nincs.) Objektumszám - 13, Súly - 5, Tárgyenergia -25. (Tehát az orvosság súlya 10, energiája 25.) A tárgyak után jöhet a "Fegyverek szerkesztése és módosítása". Ez nagyon hasonlít a tárgyak speciális tulajdonságainak beállításaira, azzal a különbséggel, hogy itt tárgyenergia helyett erő van. Ez a szám jelenti az adott fegyver erejét. A példának felhozott világomban összesen csak két fegyver van, a 11-es számú kés és a 12-es számú fejsze. Objektumszám - 11, Súly - 10, Erő - 5. Objektumszám - 12, Súly - 25, Erő - 12. Az akadályok speciális tulajdonságainak beállítása a legbonyolultabb:

Itt 11 oszlopot készítünk. Az első itt is az "Objektumszám". Annak az akadály típusú objektumnak a száma, amihez be akarjuk állítani a speciális tulajdonságait. Második a "Leírás" oszlop. Ide egy 255 karakter hosszú szöveg kerülhet. Ez közvetlenül a szoba leírása után fog megjelenni. A "Lezárt irányok" az akadály egyik legfontosabb mezője. Ide egy 12 karakter hosszú számsor kerül, aminek mindegyik karaktere vagy 0, vagy 1. A 12 karakter a 12 irányt jelzi.

Észak, északkelet, kelet, délkelet, dél délnyugat, nyugat, északnyugat, fel, le, ugrik, mászik. A 0 azt jelenti, hogy ez az irány le van zárva, az 1 azt, hogy ez az irány szabad. Tehát egy 001010001010 sor azt jelenti, hogy az akadály a keleti, déli, fel, ugrik irányokat lezárja, a többit szabadon hagyja.

A következő oszlop a "Figyelmeztetés". Ide egy tetszőleges szöveget lehet beírni. Ez a szöveg fog megjelenni akkor, amikor a játékos egy olyan irányba próbál menni, amelyet az akadály lezár.

A következő a "Feltétel".

Ide annak a tárgynak a kódja kerül, amelyik megléte szükséges az akadály legyőzéséhez. A következő 4 mező is abban az esetben hajtódik végre, ha a játékosnál nincs a feltételben meghatározott tárgy. Ezek: parancsonkénti mínusz energia - ennyi energia vonódik le a játékostól, minden kiadott parancs után, x mp.-kénti min. energia - x másodpercenként ennyi energia vonódjon le a játékostól, x mp. értéke - az előző levonás másodpercértékének beállítása, azaz ennyi másodpercenként fog levonódni. az előző mezőben megadott érték, max. parancsok száma - ennyi parancsot adhat ki maximálisan a játékos az akadály mellett, utána meghal. Mikor a játékos belép az adott helyszínre, az már 1 parancsnak számít. Tehát, ha a "max. parancsok száma" mező értéke 1, akkor a belépő játékos (hacsak nincs nála az a tárgy, amivel semlegesíteni tudja az akadályt) azonnal meghal. Ha ennek a mezőnek az értéke 2, akkor a belépő játékos még egy parancsot kiadhat. Például elmenekülhet. Ha nem ezt teszi, meghal. Ha a mező értéke 3, akkor a játékos 2 parancs kiadása után hal meg, és így tovább. A következő oszlop az "Engedély". Ennek értéke

vagy 0 vagy 1 lehet. Ha 0, akkor, ha a játékosnál van a feltételben meghatározott tárgy, akkor az akadály megjelenik (kiíródik) de a játékos hagyja szabadon mozogni, illetve nem von le az energiájából. Ha 1, akkor a játékos számára el is tűnik az akadály. Nem lesz kiírva neki. Az utolsó oszlop a "Halálüzenet". Ez az üzenet jelenik meg a játékosnak, ha az akadály megölte őt. A példa világomban 3 akadály van. Az egyik az ablak (kódja a 6-os), a másik a húsevő növény (kódja a 9-es), és a harmadik az ajtó (kódja a 18-as). Az ablakot az 5-ös kódú létra, a húsevő növényt a 10-es számú magas szárú csizma, az ajtót az 1-es számú kulcs semlegesíti. Ablak: Objektumszám - 6, Leírás - Magasan, az ajtó fölött egy nyitott ablak van., Lezárt irányok - 000000001000, Figyelmeztetés - Nagyon magasan van az ablak. Nem éred el., Feltétel - 5, Parancsonkénti min. energia - 0, X mp.-kénti min. energia - 0, X mp. értéke - 0, Max. parancsok száma - 0, Engedély - 0, Halálüzenet - . Húsevő növény: Objektumszám - 9, Leírás - A földön mindenfelé húsevő növények vannak és megpróbálnak a bokádba harapni, Lezárt irányok - 000000000000, Figyelmeztetés -, Feltétel - 10, Parancsonkénti min. energia - 0, X mp.-kénti min. energia - 5, X mp. értéke - 2, Max. parancsok száma - 0, Engedély - 0, Halálüzenet - Megettek a növények! Ajtó: Objektumszám - 18, Leírás - A bejárati ajtó zárva van., Lezárt irányok - 000010000000, Figyelmeztetés - Zárva van, kulcs nélkül nem jutsz be., Feltétel - 1, Parancsonkénti min. energia - 0, X mp.-kénti min. energia - 0, X mp. értéke - 0, Max. parancsok száma - 0, Engedély - 1, Halálüzenet - .

Objektumok elhelyezése a pályán

Miután minden objektum tulajdonsága be lett állítva, el kell őket helyezni a pályán. Ennek a táblázatnak három oszlopa lesz. Első a "Szobaszám". Ide annak a szobának a száma kerül, ahová az objektumot helyezni akarjuk. A következő az "Objektumszám". Ide annak az objektumnak a száma kerül, amit el akarunk helyezni. Végül pedig a "Darab" jön. Ez a mező mutatja, hogy a beállított szobában, a beállított objektumból hány darab van. Ez akadály esetén 1. Tárgyak fegyverek esetén -1-re érdemes beállítani. Ez azt jelenti, hogy az adott tárgyból vagy fegyverből végtelen számú van a pályán. Tehát lerakom a pályára az objektumaimat: Szobaszám - 7, Objektumszám - 1, Darab - -1, Szobaszám - 9, Objektumszám - 2, Darab - 1, Szobaszám - 8, Objektumszám - 4, Darab - 1 ... Szobaszám - 2, Objektumszám - 11, Darab - -1 ... Szobaszám - 10, Objektumszám - 13, Darab - -1 ... és így tovább.

Vizsgál parancsok leírása:

A táblázatban összesen 2 oszlop van. Az első az "Objektumszám". Ide annak az objektumnak a számát kell írni, amelyikhez szöveget szeretnénk rendelni. A "Szöveg" mezőbe pedig a vizsgálateredményeként megjelenő szöveget kell beírni. Tehát: Objektumszám - 1, Szöveg - Ez bizony pénz., Objektumszám - 2, Szöveg - Pénzért cserébe sört ad!, Objektumszám - 3, Szöveg - Ez egy üveg sör. És így tovább.

Nyom-húz kapcsolatok kitalálása:

Ez már egy kicsit bonyolultabb.

Azokat a tárgyakat kell ide felvenni, amelyek megnyomása vagy húzása esetén, valamilyen új objektumot hoznak létre vagy tüntetnek el. Nálam ilyen a fiók, amit meghúzva egy kulcsot talál a játékos. Ebben a táblázatban 7 oszlop legyen. Az első az "Objektumszám". Ez annak az objektumnak a száma, amelyikhez a nyom vagy húz parancs tulajdonságait akarjuk rendelni. A következő a "Szöveg". Ez akkor jelenik meg, ha a játékos az adott objektumra a játékos kiadta a nyom vagy húz parancsot. A következő a "Parancs" oszlop. Ide annak a szónak a parancskód mezője kerül, amelyik az objektumra kiadható. Például, ha a nyom, akkor 1, ha a húz akkor 2. Ha új szavakat vettünk fel, akkor azok parancskód mezője. (pl.: lök, teker, stb.) Az "Eltűnik" oszlopba 1 kerül, ha az objektum a kiadott parancs után eltűnik, 0 ha megmarad. Ez csak a hátizsákban lévő objektumra érvényes, a pályán lévőre nem. Például ilyen lehet egy fémcső gombbal, amit ha megnyom a játékos az lézerkarddá alakul. Ebben az esetben a fémcső objektum eltűnik a hátizsákból és egy új objektum, a lézerkard fog megjelenni. Ha egy másik objektumot akarunk eltüntetni (nem azt, amire a kiadott parancs vonatkozik), akkor a következő, a "Törölt objektum" mezőt kell használni. Ide annak az objektumnak a száma kerül, amit el akarunk tüntetni. Például egy kart meghúzva eltűnik egy rács, és így a játékos tovább tud menni. Ebben az esetben nem kell az objektumnak a hátizsákban lennie, mint előbb. A következő két mezőnek akkor van jelentősége, ha a parancs kiadása után, egy új objektum jelenik meg. Például a rács felemelkedésével kiszabadul egy oroszlán. Az első amit meg kell adni, hogy milyen objektum jelenjen meg. Az új objektum számát az "Új objektum" mezőbe kell írni. Az "Új objektum helye" mezőbe 0 vagy 1 kerülhet. Ha 0, akkor a hátizsákba kerül az új objektum, ha 1 akkor a pályára (az aktuális helyszínre). A pályára csak nem mozdítható objektum hozhat létre objektumot. Tehát felvehető objektum csak a hátizsákba hozhat létre új objektumot, azaz felvehető objektum esetén ennek a mezőnek az értéke biztos, hogy 0 kell, hogy legyen. Nekem tehát a fiók az egyetlen olyan objektumom, amelyet a nyom/húz táblázatba be kell jegyezni. Objektumszám - 7, Szöveg - A fiók kicsit akad, de azért kinyílik, és találsz benne egy kulcsot, amit rögtön magadhoz is veszel.,

Parancs - 2, Eltűnik - 0, Törölt objektum - 0, Új objektum - 8, Új objektum helye - 0. Tehát a 7-es kódú, azaz a fiók objektumra (Objektumszám - 7) kiadva a nyom/húz parancs 2-es parancskódú változatát, azaz a "húz" parancsot (Parancs - 2), a játékos hátizsákjában (Új objektum helye- 0) megjelenik a 8-as kódú objektum, azaz a kulcs (Új objektum - 8), és kiíródik neki a "Szöveg" mezőben olvasható mondat.

Használ kapcsolatok leírása:

Ez az egyik legfontosabb táblázat.

Ez tartalmazza, hogy melyik objektum, melyik objektummal használva, mit hoz létre. Ennek 6 oszlopa van. Az első kettő (Objektumszám1, Objektumszám2) azt mutatja, hogy melyik két objektum használatáról van szó. Az "Objektumszám1"mezőben lévő objektumot használva az "Objektumszám2" mezőben lévő objektummal a következő 4 mezőben meghatározott dolgok történhetnek. Ha az "Első objektum eltűnik" értéke 0, akkor az "Objektumszám1" mezőben megadott objektum eltűnik, ha 1, akkor megmarad. Ha a

"Második objektum eltűnik" értéke 0, akkor az "Objektumszám2" mezőben megadott objektum eltűnik, ha 1, akkor megmarad. Az "Új objektum kódja" mezőbe annak az objektumnak a kódja kerül, amelyik az első két mezőben meghatározott objektumok használatával létrejön. (Ha 0, akkor nincs ilyen.) Az "Új objektum helye" pedig azt mutatja, hogy hol fog létrejönni az új objektum. Ha ennek értéke 0, akkor a hátizsákba kerül, ha 1, akkor a pályára (az aktuális helyszínre). Ilyen nálam a pénz (1-es kódú) + automata (2-es kódú) =sör (3-as kódú) és a sör + festő (4-es kódú) = létra (5-ös kódú) kapcsolat. Objektumszám1 - 1, Objektumszám2 - 2, Első objektum eltűnik - 0, Második objektum eltűnik - 1, Új objektum kódja - 3, Új objektum helye - 1.Objektumszám1 - 3, Objektumszám2 - 4, Első objektum eltűnik - 0, Második objektum eltűnik - 1, Új objektum kódja - 5, Új objektum helye - 0. Tehát a pénzt használva az automatával, sör fog megjelenni a pályán, és a pénz eltűnik.A második esetben, a sört használva a festővel, létra fog megjelenni a hátizsákban, és a sör eltűnik.

Még néhány gondolat:

A példa világban csak két fegyvert helyeztem el. De mivel a világ kicsi, ezért a többjátékos funkcióra teljesen alkalmatlan. Ha nekiállunk egy nagy világ megtervezésének, ne felejtünk el különböző fegyvereket szétszórni a pályán, különben a kalandozók nem tudnak majd megküzdeni egymással. Ne felejtsük el olyan tárgyak elhelyezéséről se, amik a játékosok energiáját növelhetik sebesülés esetén. Érdekesebbé tehetjük a játékot, ha olyan tárgyakat is elhelyezünk a világunkba, amik semmire sem jók, vagyis a küldetés szempontjából teljesen érdektelenek. Ezzel megzavarhatjuk a kalandozót. Ugyanilyen lehet, ha olyan akadályokat helyezünk el, amiket nem lehet legyőzni. Például beállítunk egy őrt egy épület ajtaja elé. Persze a játékos soha nem fog bejutni az épületbe, de nem is kell neki bejutni. Sőt, igazából nem is csináltunk épületet az őr mögé, az akadály csupán egy nem létező utat zár le. De ezzel elérhetjük, hogy a kalandozó törje a fejét a nem létező megoldáson. Figyelni kell arra is, hogy semmilyen irányt nem szabad eltitkolni. Hiszen a játékos csak a mi leírásunkból ismeri meg a világot, amelyben járkal. De nem is kell feltétlenül mindig egyértelműen rámutatni a lehetséges irányokra. Ilyen például a fa az előbb megírt világban. Az van a szövegben, hogy "Egy magas fa áll a kert közepén.". Nem írtam, hogy felfele is lehet menni, viszont az, hogy a fa "magas", eszébe juttatja a játékosnak, hogy akkor bizony meg kéne mászni azt a fát.

Talán mégsem olyan egyszerű (lett volna) ezt A JÁTÉKOT elkészíteni ??? Gondolkozatok el rajta...
Üdv.Játékgyáros