

AMIGAonly

Minden ami Amiga...

2

Az AMIGAonly kiadásnapja: 1995. március 15.

Amiga Tulajdonosok lapja

Ezúton is szeretnénk üdvözölni a **Scenest** party minden résztvevőjét, sok sikert kívánva a versengő csapatok minden tagjának, és mega greetings-ünket a rendezőgárda érdemesült tagjainak...

Továbbá meg szeretnénk köszönni minden általunk ismert (és kevésbé ismert) olvasónknak az elhangzott, és-építő kritikát, a segítségfoslányokat, amik remélhetőleg a későbbiek folyamán valódi segítségé válnak.

Remélhetőleg, ezekből a kizárólag ismertető jelleggel (és nem haszonszerzés céljából!) készülő cikkekből nem kerekedik ki ismét egy országos balhé, és elkerülhető lesz a BSA-val való összeütközés is. Ugyanis egy olyan programról nem szabadna semmiféle publikációt közölni, ami nincs bizonyíthatóan a tulajdonunkban. Tehát ezúton is megkövetjük mindazon software-cégeket, akiknek bármilyen anyagi kárt okoztunk, illetve okozni fogunk működésünk folyamán.

Remélem ezek a sorok nem összeráncolt homlokokat, hanem mosolyt fakasztottak, és erősen nonprofit jellegű, leginkább kulturális erőfeszítéseink nem fogják a veszűnket okozni...

Üdvözlettel,

az **AMIGAonly** teljes jelenlegi stábjá.

Scenery Animator 2.

Újdonságok

How 2 Use a Fanatic

Virtual Reality 1.

AmigaE programozás

SuperBase4 Pro 1.

Amiga OS kezdőknek

A



INTRO

AMIGAonly = FUNzine?

**Elsősorban szeretnénk felhívni mindenkinek a figyelmét arra a tényre, amit a sok ökörség miatt mindenholnan kihagytunk, miszerint :
AZ AMIGAonly az egész világon jelenleg csak két helyen kapható:**

- ÁRAMÜTÖTT ALIGÁTOR (a WEST-END udvarban a Nyugatinál)
- A PC&Model2000 Kft (Wesselényi utca 21.)

**Azonfelül megrendelhető a Postafiókunkban is.
(1399 Budapest Pf.: 701/836)..**



Ez itten már a második szám! Az újság 25%-kal lett nagyobb, mint az előző szám, ami nem más mint az ÁFA. (inkább nem)

Újságunk egy kicsit klubújság, egy kicsit FANzine, és nagyon FUNzine.

(aki tud angolul, az érti miről van szó, ez csak azért volt szükséges, mert legalább nem vagyunk illegálisak, mint jogelődünk a Vörös Újság)

Komolyabb vizekre fuldokolva, megemlítjük, hogy mivel nem vagyunk oda a nagy lóvékért - hasznukat félredobva - ingyenesen hirdethet bárki és bármi, aki nem kalóz, nem ad pénztért olyan software-t amit nem ő írt, nem PC alaplapot akar eladni (kivételt teszünk, ha a célból adja el, mert úgy gondolja, jó lenne ágy lábának), továbbá az aki nem SEX telefont hirdet!

Cégek is térítésmentesen helyezhetnek el általuk beírt karaktereket az újság lapjain, ha az előbb említett követelményeknek maradéktalanul meg tudnak felelni.

Mi egy Non-Profit jellegű társaság vagyunk (eltekintve attól, hogy mindnyájan a Rózsadombon lakunk, és mobiltelefonokkal vezetjük a BNV-nket..).

MÉG EGY FONTOS KÖZLEMÉNY!

Ha minden igaz, az AMIGAonly 1. száma megrendelhető lesz címünkön - utánvéttel. (ha kell újság írtad meg nekünk, majd összegyűjtjük a kívánságokat, és valami majd csak lesz..)

Greetings to.:

- Louise & Magic (szokás szerint - az emberfeletti munkánkért)
- 1 db muffnak, aki mélyenszántó gondolataival új arcot kölcsönzött újságunk két oldalának (köv. számunk már NÓKLAPJAonly címen fog megjelenni)
- Frankiss kollégának, aki embertelen nyelvkörnyezetből ollózott infokkal gyarapította újságunk oldalszámát
- mAsHnek, aki ugyan igencsak szellősen írt de reméljük, hogy a közeljövőben nem fog szellősebben írni..
- továbbá a MagyarOrszágai BBS tulajdonosoknak (kiemelve a nagymultú MuffBuseters-t) akik igen nagy mértékben járulnak hozzá az AMIGA scene életbenmaradásához
- a SCENEST party-n résztvevő skacoknak, akik remélhetőleg barátilag fognak a végtermékünkhöz viszonyulni.
- és mindenkinek, aki a betűkrő raztracézódó valamikre agyukkal reagálnak...

A kommunikációs programokhoz hasonlóan a Netware, vagyis Hálózati programok is két vagy több gép közötti adatátvitelt, illetve kölcsönös hardware használatot tesznek lehetővé.

A világ kisszámítógépesei körében talán legnépszerűbb a Novell Netware program, amihez tartozik még természetesen néhány kártya is, amin keresztül az adott hálózat Server gépeinek winchester-et is használhatjuk, üzeneteket küldhetünk illetve fogadhatunk más felhasználóktól stb. Ennek a software-nek természetesen van Amigás megfelelője is, ami teljes mértékű kompatibilitást jelent az összes ilyen hálózattal. Különbségeként talán csak annyit lehet megjegyezni hogy egy pc-s EtherNet kártya kb fele annyiba kerül mint ugyanez egy A2/3/4000-esbe... (a forgalmazó szerint persze ez a kártya jóval többet tud, és nem is olyan drága) Persze léteznek olyan hálózati software-ek is amelyek nem igényelnek különösebb ráfordítást, néhány kábel vagy TTL segítségével megoldható a gépek közötti kapcsolat.

De ha az emberek meg ennél is kisebbek az igényei, akkor jön az Amiga Club-ok kedvence a ParNET... Ugyanis ehhez semmi más nem szükségeltetik, mint egy 17 eres kábel és két 25 pólusú párhuzamos csatlakozó. Ebből egy igen bonyolult hálózati 'kártyát' kell építeni, és már indulhat is az adatáramlás. Egy ilyen kis csoda nagyon hasznos lehet, ha a hön szeretett Amiga modellunktól nem kellett megválnunk az Új kedvéért, és szeretnénk mindkettőt teljes értékűként használni, nincs más hátra mint bele a Parallel-be a kábelt, a Workbench lemezre felpakolni a ParNET-et, és ha mindkét gépen ügyesen megy, akkor van két szupi wincsis meg CD-ROMos gépünk. Természetesen mivel a fejlesztők tudása soha nem merül ki, megoldották azt is hogy ne csak kettő, hanem max 254 gép lehessen ilyen módon összekapcsolva... Az ehhez szükséges interface tartalmaz 8 darab TTL IC-t és 2 darab kondenzátort. A megépítése nem igényel sem különösebb agyi, sem komolyabb anyagi ráfordítást, ami a jelenlegi magyar fizetőeszköz mértékében Úgy 2500Ft-ot tesz ki. Mivel az emberi hülyeség határtalan, meg kell itt jegyezni hogy mielőtt bármilyen dolgot bedugsz a gépedbe, ellenőrizd le inkább kétszer amit készítettél, mert ebben az esetben az egyik VIA (8520) hálálával kell számolni, aminek cseréje 7000Ft körüli összegbe kerül. Nagyon fontos, hogy az összekötő kábel bedugasakor mindkét gép ki legyen kapcsolva!

Na még mielőtt bárkinek elmenne a kedve az egészszől, ássunk egy kicsit mélyebbre. A ParNET tulajdonképpen nem más mint egy DEVICE, amin keresztül az oprendszer a másik gép drive-jait, device-szait egy 'NET:' név alatt elérhetővé teszi. Ez hasonlít ahhoz mint ahogy a lemezen elhelyezkedő könyvtárak jelennek meg ha rákattintasz az icon-jára.

A ParNET használatához szükség van a következő file-okra:

L:NetPNet-handler
DEVS:ParNET.device
L:Mouse-handler
C:NetPNet-Server

továbbá a megfelelő MountList-ekre a devs: könyvtárban.

Amikor a már felinstallált ParNet-eket használni akarjuk, el kell dönteni, hogy az összekapcsolt két gép közül melyik legyen a HOST, és melyik legyen a CLIENT... Ezzel csak azt határozzuk meg, hogy a NetPNet-Server nevű program UNIT címei melyik gépet jelölik. Vagyis a HOST gépen a NetPNet-Server UNIT 1 parancsot kell kiadni, míg a CLIENT-en a UNIT 0 fogja létrehozni a kapcsolatot. De hogy ne legyen olyan egyszerű az élet, a ParNet.device UNIT számai ennek pont a fordítottjai... Tehát miután elindítottuk a Server-t, Mount-olni kell a megfelelő MountList-ből a NET: nevű device-t. Ha ezeket mind a két gépen végrehajtottuk, és megjelenik a NET: icon-ja, rá kell döbennünk, hogy nem látjuk a másik gép egyik partícióját sem, pedig ha a NET:-re kattintunk, meg kell jelenniük... Ennek az az egyszerű oka van, hogy nem tanítottuk meg a rendszert a másik oldalon lévő device-ok felismerésére. Ha például CLI-ből beírjuk azt a sort hogy CD NET:RAM , csodák csodájára, nem ír hibát, hanem az aktuális directory a NET:RAM lesz! Itt megjelenik egy értelmes kérdés, miszerint miért nem ismeri meg automatikusan a madzag másik végén levő gép használható device-ait, mikor valójában láthatóak... Talán azért, mert ha én nem akarom, hogy a velem kapcsolatban álló user lássa az összes partícióm, akkor nem mondom meg neki hogy van, csak arról kap egy listát, amihez hozzátérhet. Ez egyfajta

védelem, amit ugyan nem túl nehéz kijátszani, (csak el kell lenni a partíciók neveit) de ha nem lenne, hiányozna. Miután megtejtettük a

ParNet-nek ezt a titkát, egyszerű a dolgunk, csinálni kell egy listát azokról a device-okról, amikhez engedélyezzük a hozzáférést, majd olyan helyre kell pakolni, ahol a másik gép látja, (erre legcélszerűbb a RAM: mert az mindig létező nev, es

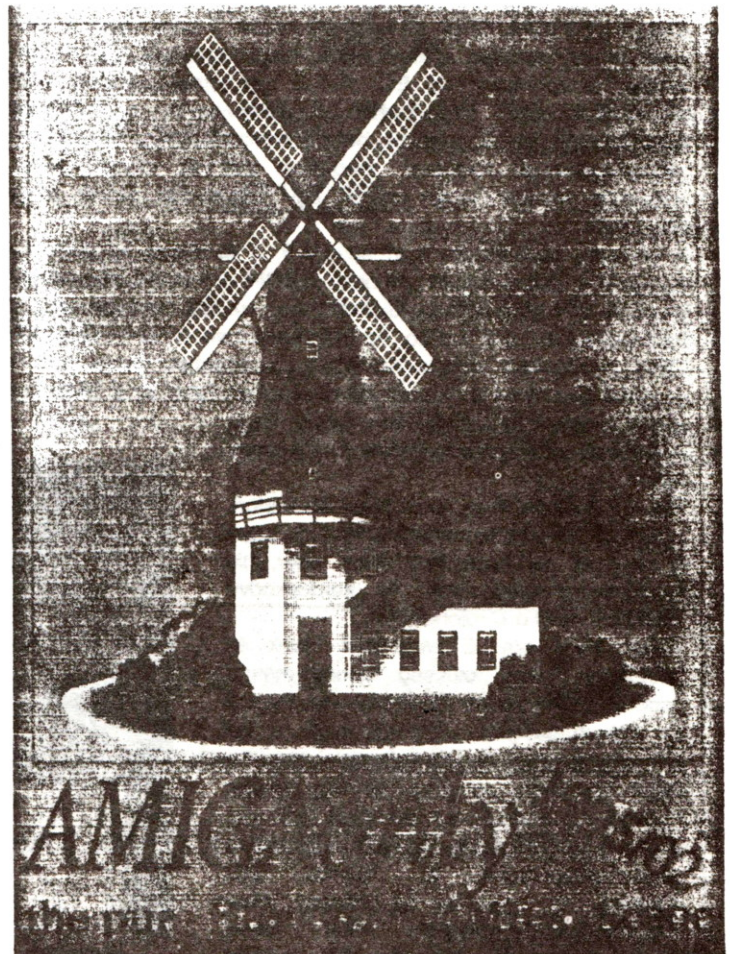
szerintem nem sok ember rejteget privát dolgokat a RamDisk-en mivel igencsak feledékeny szegény) majd egymás DriveList-jait egyszerűen Execute-áljuk, aminek az lesz a következménye, hogy diadalittasan bitet rázva örvendezhetünk a kapcsolatnak.

Biztos felőlük benned a kérdés, hogy mi a fenének kell ezzel annyit főkölni, mikor ha rendesen fel van installálva a ParNet, akkor ez automatikusan megtörténik... Hát azért, mert a tapasztalataim azt mutatják, hogy a legritkább esetben van korrekt módon, általában csak átmásolják egymástól a ParNet könyvtárat, és aztán jön a váu, hogy miért nem megy, meg hogy hogy biztos rossz a kábel, stb. Hoppá, itt vagyunk a kábelnél... Ez egy kényes dolog, pedig igazán egyszerű, mivel kell hozzá két darab 25 pólusú csatlakozó, meg egy 17 eres (lehetőleg árnyékolt) kábel. A legfontosabb dolog a bekötésnél, hogy a földpontokat (GND) hiánytalanul és hibátlanul kell összekötni, mivel ha valamelyik gép nem olyan hálózati csatlakozóban foglal helyet, ami földelt, akkor mindentéle statikus feszültségek miatt hazavághatja valamelyik, vagy mindkét 8520-at! Ha ezeket jól kötöd be, akkor szinte biztos hogy életben marad a géped. Tehát szerencsére semmi fordítás, meg csere nincs a csatlakozókon, teljesen szimmetrikusan kell felépíteni a kábel két végét. Szükség lesz a 8 darab adatvezetékre (D0-D7), ezek a 2-től 9-ig számozott lábak, az előbb említett 5 földre (GND) 18-22, továbbá a 10-től 13-ig terjedő lábkapucra. Ezek a számok nem a gépben lévő csatlakozóra, hanem az általad megvásároltra vonatkoznak, és szerepelnek is azon!

- (2-9) D7-D0 ——— D7-D0
- (12) POUT ——— POUT
- (11) BUSY ——— BUSY
- (13) SEL ——— SEL
- (10) ACK ——— ACK
- (18-22) GND ——— GND (18-22)



Louise



Az *AmigaE* története tulajdonképpen csak a számítástechnika legújabbkori történelmének része, mivel az első publikus verzió megjelenése mintegy másfél-két éve történt. Alkotója, **Wouter van Oortmerssen** megpróbálta egybefoglalni az uralkodó nyelvek (C, Modula-2, Pascal) összes jó tulajdonságát, és kipótolni azok hiányosságait, ami elsősorban az extrém magas szintű objektumorientáltság, és az igencsak nagy fordítási sebesség esetében sikerült a leglátványosabbra. Az *AmigaE*-t nevezhetnénk *E++*-nak is, mint a méltán népszerű *C++*-okat, vagy a *Pascal* objektumorientált verzióit. Azen lehetne vitatkozni, hogy a magas szintű nyelveken (*Basic*, *AMOS*, stb.) írt programok miben különböznek az *Assembly* vagy *C* nyelven megírtól, mennyiben jobbák vagy rosszabbak, de csak egyetlen mérvadó összehasonítási alap létezik: a végtermék. Nem hiszem, hogy manapság ne lehetne a MaxonBasic3-mal, vagy a BlitzBasic-kel megírni ugyanazokat a programokat, amiket jelenleg a SAS/C-vel vagy a MaxonC-vel generálnak. Itt a különbségnek leginkább az a jelentősége, hogy egy *C++* nyelven megírt algoritmus, vagy program szinte módosítás nélkül adoptálható bármely olyan platformra, ahol szintén létezik *C++* fordító. Tehát ha a programfejlesztő cég meg akarja írni mondjuk a LightWave-et más platformra, akkor egyszerűen a *C* kódot szövegfile-ban átviszi, és kis módosításokkal lefordítja a másik compiler-rel. Ugyanez mondjuk egy *AMOS* source-al igencsak nehézkesen megoldható lenne...

AmigaE

version 3.1

Tehát visszatérve az *AmigaE*-re, sajnos itt is megvan az előbb említett transzformálhatatlanság, mivel tudtommal, még más platformon nem létezik *E* fordító. De azt hiszem ezzel ki is merítettem az *AmigaE* rossz tulajdonságait, például, amikor először megláttam hogy egy 7MHz-es Amiga500-on 100 programsort 4 másodperc alatt épített össze egy futtatható file-lá, nem akartam elhinni. Pedig ez az eszemment fordítási sebesség jellemzi legjobban a fordítót. A készítője szerint több mint 20000 sort tud lefordítani 1 perc alatt! A fordítás során nincs szükség külön LINKer programra, sőt egy *Assembly* fordítót is tartalmaz az *EC*. Az *E* programba minden különös előkészület nélkül beépíthetők az *ASM* sorok, a mnemonikok használhatók mint függvények, belőlük komplett alprogramok írhatók a sebességnövelés kedvéért, megtartva azt az előnyt, hogy az *assembly* részrel együtt mégis egy magasabb szintű nyelvvel dolgozunk.

A program installációja igen egyszerű, a *Modules* könyvtárat egy *Assign*-nal *EModules*-sé, továbbá a *BIN*-t a *Path*-tal elérhetővé kell tenni. A *BIN* könyvtárban található meg az *EC* nevű program, ami tulajdonképpen az *E* fordító. Az *EModules*-ben, a file-ok hasonlóak a *C*-nél megszokott *Include* file-okhoz, de ezekben benne van az a program is aminek a függvényeit használjuk, tehát az *E* Modulokban együtt megvan az *Include* és a *Lib* is. Ez is jelentősen csökkenti a fordításhoz szükséges időt.

Nos azt hiszem itt az ideje hogy konkrétan, példákkal illusztrálva megismerkedjünk ezzel a nyelvvel, és talán lesz olyan user, aki megismerve megszereti, és kardoskodva vitatkozik majd a *C* rajongókkal...

Ahhoz hogy egy *E* programot írjunk és lefordítsuk, nincs szükség semmi másra, csak az *EC*-re és az *EModules*-re. Persze az *EModules*-ben találhatóak azok a kis programrészletek és deklarációk, amikből a majd lefordított program fel fog épülni. Ezekből az *AmigaE* 3.1 esetében 230 van, amik... tartalmazták az *AmigaOS* rendszerfüggvényeit is. Vannak más Modulok is, például a *Magic User Interface*-hez, különféle GUI típusú rendszerekhez, stb. Egy *C* fordító esetében egyszerű dolgunk van a meglévő függvényekkel kapcsolatban, mivel a megfelelő *Include* file text formátumú, kiolvashatjuk belőle a paraméterezését,

deklarációját, visszatérési adattípusát, meg esetleg valamilyen kommentet is róla. Egy *Module*-ből ez egy kicsit nehezebb, mert nem text formátumú, de az *AmigaE*-hez tartozik egy ügyes kis program, ami ezen a nehézségen átsegít minket. Ez a **ShowModule**, aminek paraméterként át kell adni a *Module* nevét, amire kíváncsiak vagyunk.

Például: **ShowModule EModules:ASL**

Ekkor meg fog jelenni az **ASL.M** nevű modulban deklarált függvények listája. Hoppá, most jut eszembe, hogy az *Amiga* user-ek szinte mindegyike nagyon utálja a paraméterezést, meg commandline írkálkakat, ezért áttérünk az *AmigaE*-hez tartozó *Editor* ismeretelésére. Ebben a jó kis editorban rádefiniálhatjuk egy billentyűkombinációra például a *ShowModule*-t és akkor

megkíméljük magunkat a CLI-zéstől...

Tehát az *AmigaE* editora az **EE**, amin első látásra pont a látványa fogott meg, sőtét háttér, nem túl világos betűk, gyorsaság, konfigurálhatóság. Ezekből is kitűnik, hogy fontos dolog, ha sok időt töltünk a monitor előtt, ne essen ki a szemünk, ne kelljen szemorvoshoz menni néhány ezer sor fehér alapon feketével való írása után! Ennek az editornak van egy külső konfiguráló programja, amin keresztül átdefiniálhatjuk a billentyűzetet, a fordító paraméterezését, stb. Van egy nagyon ügyes dolog benne, a **Complete** (jobbA és /), ami a **PROGDIR:Prefs/Completions** könyvtárban található, (és a **PROGDIR:Prefs/CompletionFiles** nevű file-ban felsorolt!) text file-okból kiegészíti az éppen elkezdett programsort.

Például: **Wri + Complete = WriteF('formatstring',args...)**

Fontos dolog, hogy az **E** szintaktikájában számítanak a kis és nagy betűk, tehát a **wri** nem ugyanaz mint a **WriteF**, és hibát fog jelezni a fordító!

Most akkor lássunk egy egyszerű **E** programot...

```
/* ProTracker Module lejátszó */
```

```
MODULE 'tools/file', 'tools/pt'
```

```
PROC main() HANDLE
```

```
DEF m,]
```

```
WriteF('Playing: ("\"), Ctrl-C to stop).\n',arg)
```

```
m,]:=readfile(arg,0,2)
```

```
pt_play(m)
```

```
REPEAT
```

```
Delay(10)
```

```
UNTIL CtrlC()
```

```
pt_stop()
```

```
ENDPROC
```



Ha olyan sorokat szeretnénk elhelyezni a programban, ami csak információ jellegű azt a /* és a */ jelek közé kell helyeznünk.

A **MODULE** egy olyan kulcsszó, aminek a segítségével használhatóvá tudjuk tenni a kívánt Modulban elhelyezkedő függvényeket (jelen esetben például a **pt**-t, amiben a *ProTracker* zenéket lejátszó

The screenshot shows the AmigaE editor environment. On the left, a code window displays the following code:

```
OPT MODULE
MODULE 'intuition/classes', 'utility/hooks', 'intuition'

/* example call:
EXPORT PROC dome
DEF h:PTR TO h
IF obj
o:=obj-SIZEO
h:=o.class
*/
```

In the background, a dialog box titled "EE Config" is open, showing various settings for the editor. The "Global Settings" tab is active, with options like "FG->BG", "Bottom Scroller", "Find Case Sensitive", "Insert Mode", "Backup", "No Fold When Loading", "Free-form Cursor", and "Fast File Loading". The "Public Screen Name" is set to "EE". Other settings include "New", "Visit", "Fallback", "Default", and "Shanghai". The "Console Spec" is "CON:0/13/675/96/EE Console,X". The "Font Name" is "e.Font" and the "Size" is "11". The "Completions" path is "PROGDIR:Prefs/CompletionFiles".

függvények vannak, `pt_play()` és `pt_stop()`).

A `PROC` egy procedúra kezdetét jelöli, a procedúrát az `ENDPROC` zárja. Minden programban kell lenni egy `main()` procedúrának, ami tulajdonképpen a főprogram, ha lefordítjuk, először ez hajtódik végre. Egy procedúrának lehetnek paraméterei, visszatérési értékei is, de erről majd később.

A `DEF` kulcsszó programváltozók DEFiniálására szolgál, alapértelmezésben mindig `LONG` típusú lesz a definiált változó. (mint a példában is...)

A `WriteF` hasonlít a `C printf`, vagy a `Basic PRINT` utasítására, egy karakterlánc vagy numerikus kifejezés megjelenítésére szolgál.

A `readfile` és a `pt` függvények a `tool/file` és a `tools/pt` modulokban található, paraméterezésükről azokból kaphatunk információt.

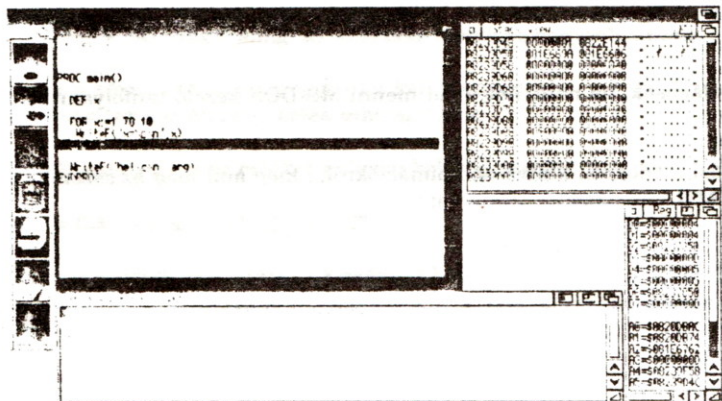
A `REPEAT` és az `UNTIL` együtt egy olyan ciklust határoz meg, ami akkor ér véget, ha az `UNTIL` után álló paraméter értéke `TRUE` lesz. Jelen esetben ha a `Control-C` billentyűkombináció lenyomásra kerül.

A `CtrlC` függvény meghívásakor leolvassa hogy meg van-e nyomva a `Control-C` billentyűkombináció, ha nincs, akkor `FALSE` állapot lesz a függvény értéke, ellenkező esetben `TRUE`.

Még az `arg`-ról nem is beszéltünk, pedig ez egy igen jelentős globális változó, amit a compiler a program paraméterek értelmezésére használ. Ez egy `STRING` típusú változó, amely space-ek nélkül tartalmazza a paramétert.

Tehát ha egy `E` programhoz hozzákezdünk, először is kell csinálnunk egy `main()` procedúrát, majd ha kidióztuk a használni kívánt modulok nevét, ezeket fel kell sorolnunk a program elején, még a `main()` előtt. Lehetőség van `Debug` információt beletördíteni a programba, aminek az esetleges hibák kinyomozásakor van jelentős szerepe. Ezt az `EC` paraméterezésében kell bekapcsolnunk... A `Debug` nem más mint a program lépésenkénti, illetve megszakításokkal tüzdelt végrehajtása. Ilyenkor van igazán lehetőség a hibák felderítésére, figyelve a deklarált változók aktuális értékét, a regiszterek tartalmát, a ciklusokban történő dolgokat. Amikor egy programot `Debug` információval fordítunk le, jóval nagyobb lesz a mérete, mert utalásokat tartalmaz az eredeti forráskód soraira. Így lehet megvalósítani, hogy a képernyőn látható forráskód soronként legyen végrehajtható. Fontos, hogy a `Debug`-ger (`EDBG`) számára elérhető helyen legyen a lefordított, és a `source` program, mert mindkettőre szüksége van a műveletekhez.

Most igencsak elnagyolva foglalkoztam az `E` nyelvvel, de ha sokkal bővebb információhoz szeretnél jutni, ajánlom figyelmedbe az `EBeginner.Guide` nevű file-t, ami `AmigaGuide` formátumban tartalmazza igen részletesen az `AmigaE`-hez vezető kezdő lépéseket.



Utóiratként, aki kedvet kapott ehhez a kultúrált programozási nyelvhez, tanulmányozgassa ezt a kis programot, mert a következő fejezetben az ebben szereplők részletezésével fogjuk kezdeni...

Sok sikert... Louise

```
/* E tiny point */
/* By EA van Breemen */
```

```
PROC main()
DEF screen,window,x,oldx,y,oldy,pendown,stoppen,code,color,lastcolor,i

screen:=OpenS(640,256,4,$8000,'E painting program')
IF screen=NIL THEN RETURN 10
window:=OpenW(0,0,640,256,8,$1000,'E Tiny point (LEFT = PD, RIGHT = PU,
MIDDLE = COLOR, LEFT+RIGHT = QUIT)',screen,15,0)
IF window=NIL THEN RETURN CloseS(screen) AND 0+10
oldx:=0
oldy:=0
pendown:=FALSE
stoppen:=FALSE
color:=1
FOR i:=0 TO 15 STEP 1 DO Box(10+(i*15),20,i*15+20,30,i)
Box(10+(color*15),20,color*15+20,30,2)
Box(10+(color*15)+1,21,color*15+19,29,color)
REPEAT
code:=Mouse()
SELECT code
CASE 1
pendown:=TRUE
x:=MouseX(window)
oldx:=x
y:=MouseY(window)
oldy:=y
CASE 2
pendown:=FALSE
CASE 4
lastcolor:=color
INC color
IF color>15 THEN color:=0
Box(10+(lastcolor*15),20,lastcolor*15+20,30,lastcolor)
Box(10+(color*15),20,color*15+20,30,2)
Box(10+(color*15)+1,21,color*15+19,29,color)
Delay(10)
CASE 3
stoppen:=TRUE
ENDSELECT
IF pendown=TRUE
Line(oldx,oldy,x,y,color)
oldx:=x
oldy:=y
x:=MouseX(window)
y:=MouseY(window)
ENDIF
UNTIL stoppen=TRUE
CloseW(window)
CloseS(screen)
ENDPROC
```

Figyelem!

Code or Die

Az AMIGAonly programozástechnikában

jártas, emberpalántákat keres,
információcsere, illetve cikkírás

céljából.

Ha érzed magadban az Erô-t ...

a címünket biztos megtalálsz valamelyik oldalon...

Nem tudom, kihevertétek-e egyáltalán a LYAPUNOV-TÉR-ről szóló leírásomat, de se baj, mert most azt is elmesélem, hogy hogyan is működik mindez gyakorlatban. Magyarán szólva most egy úgynevezett Mandelbrot-halmaz megjelenítő dologról fogok itten regélni. A programhoz van egy AmigaGuide is MandelMania.guide néven. (ha kell valakinek, szívesen elküldöm, mivel részben én magam is abból bogarásztam ki egyes dolgokat)

Akinek valamely furmányos okból nem indulna el a MandelMania 4.1 az vegyen egy (kettő) nagy levegőt, mert a program igénye kb 300 KByte chip- és kb. 700 KByte fastmem. (NoFastmem esetén gondolom kb 1 MByte, de majd Louise utánanézz...) Esetleg baj lehet, ha nincsen ASLlibrary, mert az azt hiszem kell neki. Meg a saját prefs-je, amit az icon info-ban a tooltypes-ben lehet beállítani, a varázslatos SETTINGS=prefs/MandelMania.prefs, ahol a "prefs" a könyvtárnév, tehát azt lehet módosítani.

Egy pár szót a lexikonok számára a MandelMania-ról...

- Az 1.0 volt a legelső kiadott verzió. Éljen!
- Az 1.1-ben volt először File-Requester.. Még jobb.. (azt hiszem hasznosabb volna a 4.1-ről írni. No meg a 4.0-ról, mint alapról)
- 4.0:

- a File-Requesterek átmentek a MandelMania képernyőjére
- Kijavították a cycling-et (színekörzést, vagy mit)
- tud kezelni 256-nál több színt is (AGA rulez).
- ki lett javítva néhány hiba a 68000 módban. (az FPU nem nyeste)
- Új ABOUT ablak. (ez a legfontosabb javítmány !!!)
- Kijavították a hibákat a különféle requesterekben (palette, file, screenmode) sőt, a File-Requester-t ki is cserélték egy újra.
- a palettát no meg a prefs-t lehet menteni.
- kijavították a 3.0 ARexx nem-nyeséseit. (ez már állítólag egészen jó)
- Javított zoom-kezelés.
- Másik fejléc... több információval. (fejléc - fejléc, de biztosan hasznos)
- Buffermode - az iterációkat tárolja, s ez a számolásoknál egy picit meggyorsítja a dolgokat. (láthatóan)
- Bármilyen FONT beállítható a vadiúj Font-Requester segítségével.
- Tettek belé Lyapunov-tér kiszámító algoritmust
- van benne belső 24Bites paletta !!! Yeah !!!
- CPM mód, 2- vagy 3D képek kiszámításához.
- Teljes AGA támogatás.

- a 4.1-ben javított dolgok.

- a cycle mód saját taskon belül fut, emellett AGA-s is ! Sőt be lehet állítani az inkrementáló, illetve dekrementáló értéket a gyorsabb scroll érdekében.
- a készíthető animációk 8 bitesek. (256 színűek)

(Az utolsó mondatomra visszatérve, igen lehetséges animációt készíteni, de csak némi ARexx -ben írott dolgok segítségével. Erről maj még a későbbiekben ejtek bizonyos szavakat.)

Az OPEN, SAVE, ABOUT stb utasítások gondolom mindenki számára volágosak, ha nem, hát lehet menni MS-DOS kezelő tanfolyamra. (Az ugyan nem segít, de aki még ezt sem tudja, az oda való...)

El is határoztam, hogy nem is beszélek másról csak a különböző számítási módokról, illetve az animációkról... meg ami még az eszembe jut majd. Akkor most arról mondanék egy keveset, hogy miket is tud ez a MANDELMANIA kiszámolni. Sorolom:

- Mandelbrot-halmaz
- Julia (a Mandelbrot-halmaz egy pontjából számoljuk)
- Lyapunov-tér
- Mandelbrot CPM (Continous Potential Method - ez kb. annyit tesz, hogy a lehetséges folytatásokat is figyeli. Nem igazán tudom lefordítani magyarra, majd később rátérek, hogy mire is jó valójában.)
- Julia CPM (ugyanaz)

Akkor most mindezek részletesebben. Fontos !! A matematika részeket elsősorban a Mandelmania.guide-ből vettem, igaz helyenként elláttam némi kiegészítésekkel.

Na akkor essünk neki....

A Mandelbrot-halmaz:

$$\text{Alakja } z[n+1] = z^2[n] + c$$

ahol... $z[n]$ és c eleme a komplex számok halmazának,

$$z[0] = 0 \text{ és}$$

c az úgynevezett komplex koordináta.

n a lépés száma (ahol éppen tartunk...)

(arra ugye mindenki emlékszik, hogy $z = a + ib$??)

A rekurzív (rekurzión alapuló) formát addig számolja a gép, amíg nem lesz:

$$|z|^2 > 8 \text{ vagy}$$

amíg n nagyobb nem lesz a maximum iterációknál.

A rekurzió tulajdonképpen egy algoritmus szerint ismétlődő lépésekből álló műveletsorozat, ahol az eredmény további műveletek kiindulópontjaként visszatér. (ezt azért az **Idegen Szavak Szótárából** másoltam ki biztos ami biztos alapon, mert ezt én kb két oldalban foglaltam volna össze...A "hátha valaki nem perfekt latinból" elv alapján.)

A megadható paraméterek:

(az első négy paraméter magát a **Mandelbrot-halmazt** határozza meg a képernyőn. Tehát hogy, s mint legyen, vagy ne legyen...)

X_{min} = a komplex számsíkon lévő koordináták valós részének minimum értéke. (a screen bal oldalától)
 X_{max} = a valós rész maximum értéke. (a screen jobb oldalától)
 Y_{min} = a képzetes rész minimum értéke. (a screen tetejétől)
 Y_{max} = a képzetes rész maximum értéke (a screen aljától)
(Hogy én ezt milyen faszul írtam le...)

$Iterations$ = a végrehajtandó iterációk maximum mennyisége. (lásd a **FRACTUALITY**-s irományomat...)
 $Color Divide$ = színelosztás - azoknak a sikeres iterációknak száma, amelyek ugyanazon színnel bírnak.

Ugyanaz a $z[n+1] = z^2[n] + c$, stb. összefüggés határozza meg,

csak hogy itt a c egy úgynevezett komplex konstans, tehát az értéke az egész (kiszámolt) képen mindenhol ugyanaz. Mivel magát a **Juliát** a **Mandelbrot-halmaz** egy pontjából kell kiszámolni.

A fenti paraméterek itt kiegészülnek még kettő értékkel, miszerint:

- x_{const} = valós része
- y_{const} = képzetes része az ún. Julia konstansnak

A többi marad....

CPM Mandelbrot-halmaz:

(No, most jön a java !!! Ez a kedvenc részem !!!)

Alakja szintén: $z[n+1] = z^2[n] + c$

ahol... $z[n]$ és c eleme a komplex számok halmazának,
 $z[0] = 0$ és
 c az úgynevezett komplex koordináta.

(nem kell sajnálni, mert ezt csak idemásoltam, nem írtam le kétszer..)

Eddig minden klappolt, ugye ?? Csak hogy. A rekurzív forma itt addig van számolva, míg nem történik meg, hogy:
 $|z[n]|^2 > 64$ vagy amíg az n értéke nem lesz nagyobb a **max. iterációknál**. (a későbbiekben **maxiter** néven fogom emlegetni.)

Ekkor a következő lehetséges pontot kell kiszámolni.
 $n > \text{maxiter}$ esetén ez az érték mindig 0 lesz.

A többi esetben ezt az értéket így kell kiszámolni:

$$p(z) = 0.5 * (\log(|z^2[n]|) / 2^n)$$

Ha meg akarjuk változtatni a kiszámolt kép meredekségét (ha valaki kiszámol egy Mandelbrot set CPM -t 3D-ben, akkor meglátja, hogy miről is beszélünk), alkalmazhatunk lineáris és logaritmusos skálázást.
(Hát ez egy elég furcsa mondat lett)

vagyis: $P(z) = \text{lineáris skála} * p(z)^{\text{logaritmusos skála(aditív)}}$

(Csak nehogy valamelyikőtök beküldje ezt Vágó István műsorába Grétsy tanár úrnak sok szeretettel, mert szegény menten rosszul lenne.)

A paraméterek:

(az x_{min} & max , illetve az y_{min} & max értékeket lásd fent, no meg az iterációkat úgy szintén.)

$log scale$ = a logaritmusos skála konstansa. (nohát)
 $color periods$ = 2 dimenziós megjelenítésnél használjuk, és azt mutatja, hogy 0-tól a megadott max. iterációkig hányszor használja fel a teljes palettát. Ha jó nagy számot adunk meg a iterationsnak, akkor láthatjuk, hogy a halmaz határainál olyan szép kis sokszínű műtyürrék leledzenek. Ha erre bőszen elkezdünk zoomolni, akkor megláthatjuk, hogy milyen is lenne akkor a renderelt képünk, ha nem gagyi 640*512-ben számoltuk volna ki. Hanem pl. 8192*8192. Ami - melleleg - egyelőre nem is lehetséges szegény AMIGÁnkon. (ha az ember nem rendelkezik vmely grafikus kártyával)

- lin scale* = a lineáris skála értéke (csak 3D módban)
 (Bp. scale = a Budapesti Skála értéke. (x) jó, ez gyenge volt...)
- plateau [%]* = a fennsík (plató) szintje százalékos értékben megadva.
 (megint csak azt tudom mondani, hogy aki megnézi az ügyis rögtön érteni fogja, hogy mi ez itten, mert hiába is magyaráznám el, azt senki (még én sem) érteném meg.)
- angle* = Úgy mint az előző, ez is csak 3 dimenzióban történő rendezés esetén jó valamire.
 Ha a szög = 0°, akkor pontosan szemből nézzük,
 ha a szög = 90°, akkor meg felülről.
 A többit találja ki mindenki saját maga..

Julia:

Alakja, a változatosság kedvéért:

$$z[n+1] = z[n] + c$$

ahol... $z[n]$ és c eleme a komplex számok halmazának,

$$z[0] = 0 \text{ és}$$

c az úgynevezett komplex konstans. Ami melleleg az egész képen ugyanazt az értéket képviseli. Talán azért lett konstans...

A rekurzív forma itt addig van számolva, míg nem történik meg, hogy;

$$|z[n]|^2 > 64 \text{ vagy amíg az } n \text{ értéke nem lesz nagyobb a maxiter-nél.}$$

Ekkor a következő lehetséges pontot kell kiszámolni.

$n > \text{maxiter}$ esetén ez az érték mindig 0 lesz.

A többi esetben ezt az értéket így kell kiszámolni:

$$p(z) = 0.5 * (\log(|z|^2) / 2^n)$$

Ha meg akarjuk változtatni a kiszámolt kép meredekségét, alkalmazhatunk lineáris és logaritmus skálázást.

vagyis: $P(z) = \text{lineáris skála} * p(z)$ logaritmus skála (adatok)

(mintha ezt már egyszer leírtam volna. Sebaj!! Ismétlés a tudás anyja!

Stílusosan: "Repetitio est mater studiorum")

A paramétereket nem írom le, mert teljesen megegyeznek a CPM Mandelbrot-halmaz paramétereivel, kiegészítve persze az x & y konstansokkal.

Aki nem tudja mik ezek, az figyelmetlenül olvasta el a Juliáról szóló részt.

(lehet, hogy mégsem az ismétlés a tudás anyja...?)

Lyapunovia:

No ez egy másik szép mese. Majd írok külön a LYAPUNOVIA c. programról, abban kitérek arra, hogy ez meg mi a nyü.

Most csak annyit dióhéjban, hogy alakja:

$$x[n+1] = r * x[n] * (1 - x[n])$$

$x[n]$ és r eleme a valós számok halmazának.

Az r mindig a komplex számsík egy pontjának egy x -, vagy y koordinátája, és értéke minden egyes iterációnál változhat.

A kedves felhasználó, vagy programozó (mindenki vegye fel a saját ingét) nevezheti például "A"-nak az x -, "B"-nek az y koordinátát. Ebben az esetben az "AB" azt fogja jelenteni, hogy az r megváltoztatja az x - és az y koordinátát is bármelyik iteráció esetén.

Amennyiben az a folyamat eredménye önmagába tartó, ennek a rekurzív formának szüksége van némi időre a feladat végrehajtásához.

A Lyapunov kitevőt a következőképpen tudjuk kiszámolni:

$$L(x) = (\log_2 |(r - 2 * r * x[n=1...maxiter])|) \text{ átlagával}$$

0 és a legkisebb Lyapunov kitevő között lévő összes kitevőt meg kell jeleníteni. (s lön...)

Paraméterek:

Az első négy paraméter itt is azt mutatja, hogy a **Lyapunov-tér** hol helyezkedjen el a képernyőn. (x, y - min, max)

lyaexpmin = a legkisebb Lyapunovia kitevő, amit még megjelenítünk.
sequence = a "Lyapunov sorrend" ("A" és "B" sorrendje) pl.: ABABABABAB, vagy AAAAABBBBB

settle = az iterációk száma a végrehajtáshoz.
iterations = az iterációk száma a **Lyapunov** kitevő kiszámításához. Értékének nagyobbak kell lennie, mint ahány színű a render screen.

Így látom ezt a cikket sem sikerült lapzártáig befejezni, de senki ne csüggedjen el, így legalább lesz második rész is. Amennyiben a következő szám nem ilyen viharos körülmények között fog készülni, akkor lehetséges, hogy e cikk befejezése mellett még írok a PRACTUALITY-ról, vagy a LYAPUNOVIA-ról. (vagy valami egészen másról...)

Magic

HunAMI Információk

PC&Modell 2000	267 9038	hardware
Amiga Service & Hardware	276 0116	rendelés
Amiga Service	153 0237	service
Amiga Service 1999 Tecnic	184 8845	rendelés
Thalypex	141 4414	hardware, software
Teletech	06 26 334 112	monitor
Derko	291 2318	service, hardware
Inter Central	173 1783	Commodore alkatrész, rendelés
Amiga Service	175 1024	service



HirBoard	215 2175	Munchausen
Local Graveyard	202 7423	Absolute!
Number of the Beast	291 4029	Lord/Absolute!
Myth	173 0552	Pogi
In The Kitchen	186 6327	Mucsi/Muffbusters
Amiga Dreamland	138 0446	Thomas Malmarits
Rejtelmes Sziget #1	06 99 332 623	Zozo
Rejtelmes Sziget #2	06 99 333 662	Zozo
Sly Trace	06 72 333 343	Sly
Toni	06 73 311 486	Toni/Triad
Magic Carpet	06 99 313 662	

Sponsored by *ComputerMagazin*

BBS

Magyarországi AMIGA BBSek

Ezek a szolgáltatások zömmel információ jelleggel bírnak!

Virtual Reality 1.

Üdvözlét minden virtuális világ iránt érdeklődő egyénnek. Komoly gondban vagyok, hogy miként kezdjem ennek a remélhetőleg cikksorozattá váló résznek a bevezetését. A legjobb lesz, ha rögtön belevágunk a közepébe.

Következzen néhány téma amivel foglalkozni szeretnék a közeljövőben:

- a Virtual Reality (VR) története
- a VR alkalmazási területei
- a VR hardware igénye
- a VR elmélet alapjai
- a 3D Stereo megjelenítés elmélete
- a 3D Stereo megjelenítés megvalósítása Amigán
- a 3D Sound elmélete és megvalósítása

Természetesen ezek csak támpontok, ha más érdekel, vagy nem írok olyan dolgokról, ami szerintetek fontos, akkor rogadjatok szövegszerkesztőt és írjatok az újság címére!

Aki komolyabban érdeklődik a téma iránt, valamint a zsebe sem kong az ürességtől, annak a figyelmébe szeretném ajánlani az AP Professional kiadványait, amelyek közül én is felhasználok többet a cikk írásához. Ezek a könyvek rengetek témakört felelnek mint például multimédia, grafika (ray-tracing), programozási nyelvek (sajnos nem Amiga-s), hálózati kommunikáció (network communication). A könyvekben gyakran lemezmelléklet, vagy CD is helyet kapott, amin a könyv programjai egytől-egyig megtalálhatók.

Ízelítőként néhány kiadvány, ami kapcsolódik a témánkhoz:

Texturing and Modeling:

Matematikai texturák, gázok modellezése, Noise texturák alkalmazása, kontúr elmosás (antialiasing)

Megjelenés: 1994 November

350 oldal, 3.5"-os lemez

Ára: £34.95

3D Sound for Virtual Reality and Multimedia Applications:

Térbeli hallás elmélete, kommunikációs rendszerek, számítógépes zene, stb.

Megjelenés: 1994 November

290 oldal

Ára: £34.95

Virtual Reality Applications and Explorations:

Kezdők és haladók számára íródott könyv a VR alkalmazásokról és azok felhasználásáról

Megjelenés: 1993

245 oldal

Ára: £29.95

Virtual Reality ExCursions:

Kezdőktől a profiig mindenki talál megfelelő információt ebben a könyvben. Jelenleg én is ezt használom a cikk írásához forrásként. Komplet példaprogramok, azok C forrassai, valamint az ezekhez kapcsolódó elmélet található meg ebben a könyvben. Mindenkinek csak ajánlani tudom. Egyetlen visszatartó erő, hogy rögtön megvásárolod az ára, bár egy megszálott VR rajongó inkább nem eszik, de...

Megjelenés: 1994 Április

503 oldal, 3.5"-os lemez, 3D Stereo szemüveg

Ára: £31.00

A fenti kiadványokat itthon is be lehet szerezni forintért, picivel magasabb áron, mint az eredeti. Természetesen, hogy hol, azt nem írom le, mert ingyen reklám még nálunk sincs, viszont bárkinek segítsek a beszerzésben, aki feldob egy levelet a postafiók címünkre. Mára ennyit a jövő nagy lehetőségének kikiáltott Virtuális Valóságról.

A következő alkalommal már komolyabban belemerülünk a VR fitokzatos tengerébe. Várom leveleiteket!

- mAsH -

Figyelem! feladat...

Tedd le a joystick-ot és vedd elő az Imagine-t!

Amigós be'ém up fejlesztéshez ray-tracer-t keresünk. A játék demo verziója rövidesen elkészül (90%-a kész), amihez a segítségedre lenne szükségünk. Amennyiben érdekel a dolog, akkor küldj referencia anyagot az újság címére (a borítékra írd rá, hogy DEEPCORE). A képeken csak saját készítésű object-ek legyenek!!!

Elsősorban fővárosi vagy Budapest közelében lakók jelentkezését várjuk, de ha színvonalas amit csinálsz lehetsz akár moszkvai is (telefon és saját repülőgép nem hátrány).

AMIGAonly
Budapest 1399 Pf. 701/836



AMIGA DOS

kezdőknek...

Talán sok user-nek feleslegesnek tűnő témát kezdek most el, mivel szerencsére egyre többen tekintik természetesnek az AmigaOS használatát. A jó öreg A500-asnak elsősorban a játékgép szerepe jutott, ezért nem igazán volt jellemző, hogy valakinél a Workbench uralta volna a monitor jelentős hányadát, mivel egy játékhoz a legritkább esetben volt szükség rá. De miután megjelent az AmigaOS 2.04 nem sokkal utána a 3.0, amelyek már normálisan kezelték a winchestereket (az A500-as HDD-k nem igazán voltak sikeresek), és sokan rájöttek hogy nem is olyan rossz használni a jó öreg OS-t, és modjuk akár egy Icon-t is hozzárendelhetünk egy játékhoz, nem kell mindenféle BootMenu meg MenuMaster képernyővel indulni. Sajnos az operációs rendszerek régebbi verziói (lásd 1.2, 1.3) igencsak lassúak voltak, köszönhetően az OFS-nek, és a később teljesen átfírt multitask kezelő függvényeknek.

Hoppá, itt már elég sok olyan információ hangzott el, amit meg kell magyaráznom, mivel most az igencsak kezdőknek szól a fáma. Először is meg kellene magyaráznom, hogy mit is jelent az Operációs Rendszer kifejezés. Tegyük fel, hogy adott egy olyan gép, ami tud összeadni, kivonni, és még néhány egyszerű műveletet egész számokkal. Na ez sem egy kutya egyszerű feladat, mivel nem tudom a tudomására adni annak a gépnek, hogy mivel mit csináljon, mert nincs vele semmiféle kommunikációs felületem. Ez a felület lehet egy lyukszállag, vagy egy írógép billentyűzet, (mint a régi szép ENIAC-os időkben) vagy lehet fényceruza, egér, hangdigitalizáló, vagy a leginkább elterjedt billentyűzet. Persze egy ilyen hardware-t valahogy össze is kell kapcsolni azzal a nagyon egyszerű géppel, mert ha csak melléteszem, attól nem fognak szót érteni. Ha ez is sikeresen meg lett oldva, szükség van egy olyan felületre is, ahol a válaszinformációkat megkaphatjuk. Itt szintén elég sokféle platform létezik, de most csak a monitorral érdemes foglalkozni. Tehát a gép elkészült a feladattal, és szeretné közölni velünk a végeredményt, aztán bele fog örülni magába, mert sokkal bonyolultabb feladat számára egy adott információ kiírása a képernyőre, mint akár számmilliószor összeadni meg kivonni. Szóval a választ átfordítja olyan alakra, hogy azt a képernyő kezelő hardware elfogadja, majd a képernyőkezelő szintén tovább konvertálja, hogy maga a monitor meg tudja jeleníteni. Szóval ezek az úgynevezett hardware eszközök alkotják a kész számítógépet. De valahogy meg kell oldani azt is, hogy ezek a teljesen különböző hardware-ek kommunikálni tudjanak egymással, azért, hogy mi meg kommunikálhassunk a géppel... Itt jön a software, ami tulajdonképpen egy program, ami tudja a gépben lévő összes chip minden titkát, (amit az Amiga esetében a fejlesztők igencsak titkosan kezelnek mind a mai napig!) és kapcsolatot teremt közöttük. Ezt a programot hívják Operációs Rendszernek, mivel a gép működtetését végzi. Az operációs rendszerek jelenleg igencsak lemezhez kötöttek, ezért a DOS, vagyis Disk Operating System kifejezés a használatos. Itt kell megjegyezni, -amit nem nagyon szoktak megérteni a kezdő user-ek- hogy a lemez, (vagy winchester, vagy nemfelejtő RAM stb) NEM azonos a memóriával, tehát tulajdonképpen a lemez csak háttértárolóként szolgál, a mérete semmilyen kapcsolatban nem áll a számítógéppel, az operációs rendszer a lemeztől olvassa be a programot a memóriába, és csak onnan tudja futtatni. Persze fontos hogy legyen háttértároló a gépünkben, de mindegy hogy az egy floppy, vagy egy 4 darab 1 Gigás wincsi, ha nincs elég memóriád, se így, se úgy nem fog futni a program... Tehát röviden: az Operációs Rendszer kezeli a hardware-t, a memóriát, futtatja a programokat, használhatóvá teszi a háttértárolón (lemezen) az adatokat (file-okat), és bitosítja az oda-vissza kommunikációt a számítógép, és a felhasználó között. Volt szó a MultiTask-ról is, ami az AmigaOS (Amiga Operációs Rendszer) egyik legkellemesebb tulajdonsága. Hogy megértsük

a MultiTask lényegét, meg kellene értenünk az operációs rendszer legbonyolultabb részét, az időosztásos processzormunkát, de azt hiszem ezt majd inkább egy későbbi alkalommal fogom részletezni, mivel akiknek ez a cikk elsősorban íródik, egy icipici kukkot sem értenének belőle, és nem is olvasnák tovább... Szóval a MultiTask egy olyan rendszer, ami lehetővé teszi több (gyakorlatilag csak a memória mérete szab határt a futó programok számának) program (Task) egyidejű futtatását. Persze ez nem egyidejű a valóságban, csak mivel az oprendszer elég jól meg van írva, és a processzorban is van egy kis szuftla, így teljesen valóságnak tűnik. (Ha elrettentő példát akarsz látni, nézz meg egy wndws-t!) Tehát ha például játszol az UFO-val, és az Imagine éppen renderel egy képet, még nyugodtan mehet egy Term, amivel valamelyik BBS-ről töltögeteg az anyagot... Ilyenkor, ha megkukkolsz a rendszer állapotát (például az RSys-szel) meglepődve tapasztalhatod, hogy majdnem 30 task fut egyszerre! (Persze ebben benne vannak az OS saját programjai is, mint billentyűzet, hang, kép stb kezelők.) Hát a Workbench-et használod, igen egyszerű elindítani a programokat, csak rá kell kattítani, és máris indítható a következőt. De ha jobban szeretsz gépelni, és a CLI-t (Command Line Interface - parancs értelmező) részesíted előnyben, illik a RUN paranccsal elindítani a programokat, mert nem mindegyik teszi magát olyan állapotba, hogy az elindítás után az a CLI amiből indítottad felszabadulhasson.

Néhány szót érdemes említeni a CLI-ről vagy más néven SHELL-ről.

Ez mint a nevében is benne van egy parancs (a parancs mindig egy program neve, mivel a számítógép csak programokat tud futtatni) értelmező, ami a billentyűzetről érkező utasításokat hajtja végre. Egy CLI-ből akárhány újabb CLI-t elindíthatunk a NEWCLI, vagy NEWSHELL paranccsal, ami azt jelenti hogy egy task-ból kiindulva, mindegyik task teljesen függetlenül képes akármit futtatni. A rendszer C:-nek Assign-olt könyvtárában helyezkednek el azok a programok, amiket direkt ilyen begépelendőnek készítettek. Ezekről majd később részletesebben írni fogunk...

Sajnos néha megrökönyödve hallom, hogy olyan ember, akinek már 3 éve van Amigája egy kukkot sem tud arról, hogy a lemezen vannak file-ok, könyvtárak, lemeznevek, boot-sector stb. Persze ez az elkényelmesedésből meg az onlyGame-ből adódik, mert ugyan ki a fene kíváncsi arra hogy mi is van a lemezen, mikor berakom aztán már ott is van a játék. Igaza van, mert aki játszik, annak ez éppen így tökéletes, minek kelljen neki ezer dolgot tudni a lemezszerkezetéről meg file-okról mikor ő csak szórakozni akar. Vissza az elmélethez... Szóval a file-okról igen sokat lehetne mesélni, de megpróbálom röviden és lényegretörően megoldani. Tehát, a lemezen rengeteg adat van, mindenféle szöveg, kép, zene stb. Ezeket valahogy olyan állapotba kell hozni, hogy mindenki által használhatóvá váljanak. Erre a legegyszerűbb módszer, ha az adatsoportoknak egy nevet adunk, amivel hivatkozni lehet arra a csoportra. Így a lemezen elhelyezkedő rengeteg információból csak nevek látszanak, és nem értelmetlennek tűnő számok, leegyszerűsítve a hozzáférést is. Tehát ha te hivatkozol egy névre (amihez például egy futtatható program sok-sok információja van hozzárendelve) az operációs rendszer tudja hogy honnan kell betölteni a lemeztől (és milyen sorrendben) az adatokat ahhoz, hogy elindíthassa azt a programot. Ha kiadod a CLI-ből azt a parancsot hogy LIST, akkor kiírja a képernyőre a lemezen található file-ok listáját, plusz még egy-két információt a file-ról:

FileNév	FileHossz	ProtBit	Készítés ideje	Komment
ForceVGA	4220	---rwd	24-Jun-93 08:06:32	VGA monitor esetén ő állítja át a képet
ForceVGA.info	468	---rw-d	24-Jun-93 08:06:33	ikon
10 files - 90 blocks used				

Tehát egy file-nak a nevében kívül van hossza, vannak olyan bitjei, amik megmondják hogy olvasható, írható, futtatható, törölhető, tartalmazza a létrejöttének dátumát, és egy nagyon okos dolog, van kommentje is. A komment egy információ amit a file-hoz fűzve tárol a

ndszer, amivel megjegyzéseket fűzhetünk hozzá. (például ha egy module-ról tudjuk hogy ki készítette, a kommentjébe beleírhatjuk, de rsze a neve nem fog megváltozni.) A file hosszát úgy kell elképzelni, mintha egy könyvről lenne szó, és a hossz a könyvben található tük számát mondja meg. Tehát ha mondjuk a Lítíl Devil-ról tudjuk hogy körülbelül 32 MegaByte, akkor az azt jelenti, hogy a játék ssza megfelel egy olyan könyvnek, amiben 32 millió betű van...

ivel az Amiga operációs rendszere (vagyis nagy része) a ROM-ban vagyis a 'csak olvasható memóriában' helyezkedik el, tulajdonképpen hoz hogy elinduljon a gép semmire sincs szükség, csak egy olyan lemezre, amit formázva (FORMAT) és installálva (INSTALL) van. A ORMAT egy olyan program, ami a lemezen (ez lehet floppy, winchester vagy RAM) található 'adathordozó réteget' olyan információkkal tja el, amit az operációs rendszer értelmezni és használni tud, tehát adatokat (programokat, képeket, zenéket stb.) tud ráírni, és onnan olvasni. Szóval ha a boltban vásárolt floppy-t, vagy winchestert nem tudod használni, annak az az oka, hogy a rajta lévő információ nincs mába öntve, vagyis még használhatatlan a gép számára. Itt külön kell választani a floppy-kat és a winchestereket, mert a floppy tében sokkal egyszerűbb dolog a használhatóvá tétel, minden előzetes HDToolBox-ozás nélkül formázható és installálható. Tehát ha zzzájutottunk egy doboz friss floppy-hoz, amire fontos dolgokat szeretnénk biggyeztetni, próbálkozzunk a formázással. A legegyszerűbb, t betesszük a lemezt, rákattintunk a Workbench ICONS menüjében a FORMAT DISK-re, és értelemszerűen kitöltjük a megjelenő LI-t... A másik megoldás, amiből sokkal többet okulhatunk, ha ugyanezt CLI-ből oldjuk meg, koptatva a billentyűket, és az ujjainkat;

FORMAT DRIVE df0: NAME üres FFS INTERNATIONAL

ppá, de honnan a fenéből tudjam, hogy mit kell a FORMAT után írni? Na itt jön a gógyi, mert a fejlesztők gondoltak ám a halandókra is, gysis ránk) és nem feltételezik mindenkiről, hogy részt vett az AmigaOS fejlesztésében, ahol megtanulhatta az utasítások raméterezését... Tehát ha úgy érzed most vége a világnak mert lebógsz a barátod előtt, akinek csak 1 file-ra lenne szüksége, de olyan mezt hozott ami még szűz, nincs más hátra, üsd be: **FORMAT ?**

enkor nem kell megilyedni, hogy hazavágja valamelyik lemezt, mert a kérdőjel a parancs után azt jelenti, hogy információt kérsz, mert m tudod a megfelelő paramétereket. A válaszban szerepel az összes használható paraméter, továbbá az, hogy a kiválasztott raméternek milyen formátumúnak kell lennie. (pld.: /S)

ldául, ha azt látod hogy

- ,OFS/S,---** az azt jelenti, hogy az 'OFS' szöveget kell kiírni (persze csak ha szükség van a szolgáltatára) minden további nélkül, tehát ez egy egyszerű kapcsoló, ami ha leírod akkor ON állásban van.
- ,NAME/K,---** ell a NAME kapcsoló után egy nevet is írni (pld.: NAME üres)
- ,DIR/M,---** a DIR helyére egy szöveg kerüljön (pld.: LIST dh0:system)

a Workbench lemezen lévő programok mindegyikére érvényes, és persze a kultúráltabb programokra is. Nos, sikeresen túljutottunk a mába öntés komoly feladatán, megtörölhetjük izzadt homlokunkat, felhívhatjuk minden ismerősünket, esetelve maximális emítástechnikai képességeinket... De még ne! Inkább installáljuk azt a szerencsétlen floppyt, mert szeretnénk rá felvenni egy komplett ndszert, ami feleslegessé teszi a már unalmas winchesterről való boot-olást!

INSTALL df0: FFS

azt jelenti hogy a lemez képes elindítani az operációs rendszer CLI-jét, annélkül hogy egy file-t is rámásolnánk.

Folytatása következik....

Louise

továbbiakban is a magunk feje után fogunk menni, de ha valakinek valami operációs rendszerrel kapcsolatos kérdése van, akkor az agodtan írhat egy (vagy több) levelet nekünk, esetleg meg is fogjuk válaszolni.

Következő részben átveszem a lantot Louise-től, és én fogom folytatni az AMIGA OS ismertetését, azt még nem tudom, hogy milyen módon, majd kiötölök valamit

Elig is mindenki bőszen gyakoroljon otthon...

Magic

SuperBase4

Az AdatBázisoló

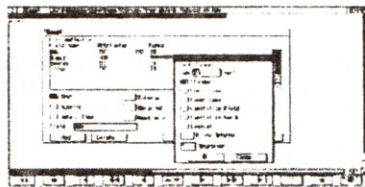
A SuperBase4 Pro a jelenlegi legfejlettebb verziója a SuperBase adatbáziskezelő rendszerek. Már több mint 10 éves múltra tekinthet vissza, és szinte minden platformon megtalálható egy-egy megfelelője. Még a 8 bites korszakban C64-re is megírták, még hozzá olyan minőségben, hogy az akkori (és ha belegondolunk a C64 adottságaiba, még a mai értékrendben is) kisgépes adatbáziskezelők legjobbjává vált. Hol van már a 64kByte meg a 160k floppy ideje... Jelenleg a SuperBase4 Pro-nak legalább 1Mega memóriára van szüksége ahhoz hogy elinduljon. De ha a Form Designer-t is használni akarjuk vele párhuzamosan, akkor illik legalább 2-3 Mega. A winchester nem alapkövetelmény, de ha belegondolunk, hogy az adatbázisok nem a kis terjedelemtől híresek, akkor a 880k floppy-kapacitás nem igen járható út. Tehát ha van kb 2Mega RAM a gépünkben, nincs más hátra, mint hogy elindítsuk az AREXX interface-t, (ugyanis ezen keresztül kommunikál egymással a Form Designer és az SBPro) aztán a SuperBase4-et. Ekkor nagy megrökönyödésünkre nem egy fekete alapon fehér betűhalmaz jelenik meg, hanem kultúrrendszerhez méltóan, egy kezelői felület. Ez alapvetően két nagy részt jelent, a felső ablakot, amely az aktuális OUTPUT, és az alul elhelyezkedő ugynevezett CONTROLL-PANEL-t.

Aki semmiféle jártassággal nem rendelkezik az adatbáziskezelésben, annak igencsak fel kell kötni a kötnivalóját, mivel a dokumentáció eleve feltételezi ennek meglétét. Tehát, ha az ember kicsi fiacskója mondjuk az otthon tárolt (persze kizárólag Shareware...) programjait szeretné valami rendszerben átlátni, szüksége lesz egy füzetre, vagy éppen egy adatbáziskezelő programra. Ha egy füzeten akarja tárolni a programjainak minden lényeges adatát, logikusan valamilyen rendszer alapján rendszereznie kell. Ez lehet a címe, szerzője, hossza, típusa, vagy éppen egyiksem... De ha mégis választott egyet, akkor van egy kiindulási alap. Tegyük fel, hogy Móríczkánk az értelmesebbik fajta képviselője, és választott. Még hozzá a program neve alapján szeretné rendszerezni a rengeteg sok természetesen nem lopott, meg jogtisztá, meg megvásárolt programjait. Ebben az esetben ilyen rajzot készít a füzetébe:

Név	Hossz	Szerző	Típus
AlienBreed	500000	Team17	Gyilok game
ArcadePool	800000	Team17	Billiárd game

Felbátorodva ezen nagyszerű feljesítményén, megpróbálkozni ugyanezt a magasröptű elméletet adóptálni SuperBase4 környezetbe. Tehát (persze csak ha már betöltötte az SBPro-t) kiválasztja a Project menüben, a New almenü, File pontját. (Ezt ezentúl Project/New/File formában fogom írni, mert nem akarom teljesen elkoptatni a mutatóujjamat...)

Itt a program megkérdezi, hogy mi legyen az új adatbázis neve, legyen mondjuk 'Programjaim'. A következő kérdése arra irányul, hogy akarjuk-e jelszóval védeni a szuperitkos adatbázisunkkal való műveleteket, úgymint Törles, Olvasás-Írás, Csak Olvasható. Ha itt nem írunk sehova semmit, akkor értelem szerűleg minden műveletet végrehajthatunk a kész adatbázissal. Most kezdődik az igazi része a dolognak, mert létre kell hoznunk a



a gyakorlatban úgy néz ki, hogy beírjuk a Mező nevét (például 'Név') aztán NEM Enter-t nyomunk, hanem kiválasztjuk a típusát (most a Text), beírjuk a hosszát, stb., majd megnyomjuk az Add button-t. Az itt található többi kis kapcsolóval igyekszem nem foglalkozni, aki a későbbiek folyamán használni fogja a programot, az majd úgyis rájön a funkciókra... Ha ilyen módon beírtuk mind a négy Mező-t, lezárásképpen az Ok button-t ajánlanám. Mint ahogy az előbbiekben már szó volt róla, kell választani legalább egy Index-et, ami alapján rendezésre kerülnek az adatok. Ha ezen is sikeresen átjutottunk, megjelenik a négy Mező, várva, hogy adatokkal töltsük fel. Ezt a Record/New segítségével kezdhethjük el, és addig folytathatjuk, amíg van mit rögzíteni. Tehát megismerkedtünk az adatbáziskezelők három legfontosabb szavával, a Field-el, a Record-al és az Index-el. Egy Record-nak vannak Field-jei, és a Record-ok Index-elve vannak bizonyos Field-ek szerint... Nem olyan bonyolult ez, csak idegesítés céljából összegeztem így, mint a régi szép tanulmányokkal eltöltött évek alatt, a tankönyvekben.

Ha ezeken az alapokon sikerül elindulni, a többi már adódik, mert szükség lesz rá.

Példul egy ilyen adatbáziskezelő, ha már mondjuk 5 millió record-ból álló adatbázissal dolgozunk, akkor is néhány másodperc alatt előkeresi a kívánt adatot, mivel az Index file-ban minden felvett Record-hoz tartozik bizonyos mutató, hogy ha szükség van rá, ne kelljen végignyálgazni az egész adatbázist, hanem egyszerűen oda ugorva, megmutathassa. Lehetőség van különféle listákat készíteni, szűrőket alkalmazni, stb. Természetesen ezek mind elérhetőek menüpontokon keresztül, de akár programból is. Ugyanis a Superbase4-nek van egy ugynevezett DML-je, amiben BASIC programokat lehet írni, kihasználva az adatbáziskezelő utasításait. De a DML programozására majd később kerítünk sort, mert ahhoz egy hosszabb lélegzetű memory-chunk kell. Visszatérve az ismertetésre, a CONTROLL-PANEL-en keresztül elérhetőek a Record műveletek, a keresés, a képek kü/be kapcsolása, a FILTER, és a programmegszakítás. Ez programból a PANEL OFF utasítással kikapcsolható, de a kurzor billentyűk ilyenkor is aktívak maradnak. A SuperBase4 képes IFF képeket is használni és megjeleníteni, tehát készíthető vele képi-adatbázis is, persze csak ha rengeteg helyünk van, mert ugyebár az IFF file-formátum nem a legtömöröbbségek közé tartozik...

Első nekifutásra, és mindent összevetve a SuperBase4 egy igen kiforrott, rengeteg adatbáziskezelő utasítással rendelkező, és nagyon megbízható program. Ha a megfelelő méretű memória rendelkezésére áll, szinte teljesen kizárt a fagyás lehetősége, kivéve, ha valami röptömörítőt (az én esetemben DiskExpander-t) használunk azon a partíción, ahol az adatbázisunk található. Ilyenkor kb tízedére leesik a Record-műveletek sebessége, és nagy Record-szám esetén a kedves Guru típusú Meditáláson vehetünk részt. De ezt igen egyszerű kiküszöbölni, mert amíg dolgozunk az adatbázison, ki kell kapcsolni a tömörítést... Legközelebb belemászunk a DML, vagyis a SuperBase4 programozásába, megpróbálunk valami fenomenálisat alkotni a kedves Mórícka segítségével...

Louise

Animations

Scenery Animator 4.0 - Natural Graphics 1993

part 2.

Miután mindenki kellőképpen lerágta a körmeit, akkor talán jöjjön a folytatás. Maga a történet ott kezdődik, ahol az előző résznek vége volt, ugyanis a két részt jóformán egyszerre írtam meg, csak később döntöttünk úgy, hogy kettévágjuk, mivel enyhén bőbeszédű lett ez a helpescke.

Visszatérvén a főmenübe a gép számolgat egy darabig, majd a preview ablakba kirakja, hogyan is nézhet majd ki a dolog. No nem kell beszarni, azért ennél majd szebb lesz. Itt is lehet még a magasságon állítani (**UP-DOWN**), amellet előre illetve hátra mozgatni a kamerát (**FORWARD-BACKWARD**), nagyítani (a **ZOOM** felirat melletti sávban - illetve magán a preview képen is, kijelöléssel ugyanis ez a kettő dolog ugyanaz. Ha elcsesztük a képünket, mert beclickeltünk, akkor kell a **ZOOM**-nak visszaállítani a nagyítást. Ez a nagyítás, nem a térképen lévő nagyítás !!!) A főmenüben inkább állítsuk be a fényviszonyokat

- **LIGHT**-, ha túl sötét a képünk (nem csak a mienk, hanem a preview is), akkor itt tudjuk beállítani - a kör és a benne lévő kis pont segítségével -, hogy merről süssön a nap, és milyen erővel. (a *Süss fel nap - Fényes nap* c. gyermekdal itt nem igazán hoz eredményt)

- A **WATER** (víz) a legjobb opció, ugyanis itt tudjuk elárasztani a nehéz munkával megkreált tájképünket. Ezért célszerű ezt az objectek elhelyezése előtt beállítani. **WATER LEVEL** - vízszint, ez egyértelmű; ha magasabb, mit az egész táj magassága az nagyon jó, mert játszhatunk Atlantisztt, vagy ha magasabb, mint a kamera, megtudhatjuk, hogy milyen is Cousteou kapitánynak, azonban értelme egyiknek sincsen. Az **OCEANS** azért jó, mert más, mint a tó, és ha ez be van kapcsolva, akkor az olyan tó, ami kimegy a térképről, óceánná alakul (valamivel setétebb lesz a színe). A **WAVES** itt fodrokat jelent. (legyen bekapcsolva, mert amúgy is eléggé csúnya ez a víz, hát még anélkül....) Visszatérvén a menübe megláthatjuk milyen pusztítást vitt véghez a víz. (a víz itt világosabb színnel van jelölve) Ha a képen csak egy fél fa van például, akkor inkább a vízszintet csökkentjük, mert az némileg egyszerűbb. (ámbár akkor csinálhatunk félfamentes rajzlapot is....) Ja, ehhez még annyi, hogy a térképen beállított **LAKE** az itt lévő értékeket nem módosítja!

- Ha minden nyesi, akkor beállíthatuk, a **LAND** gombbal, hogy hogy is nézzen ki a dolog.

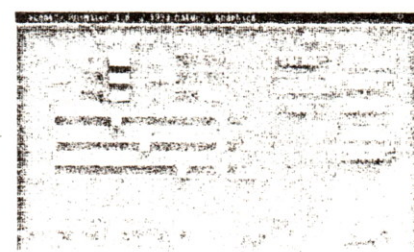
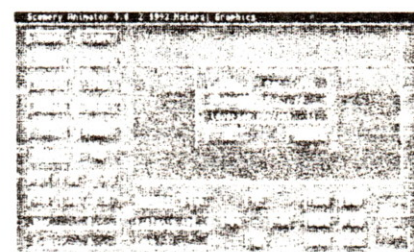
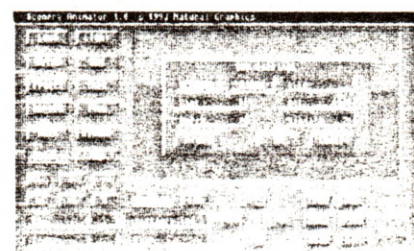
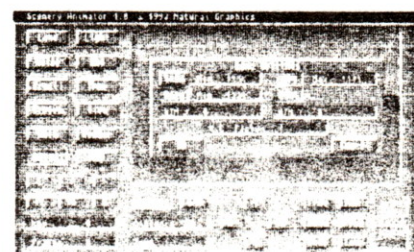
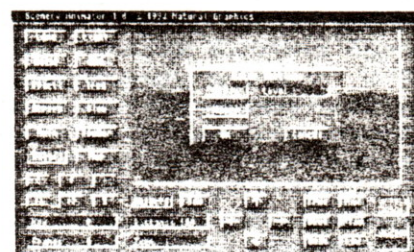
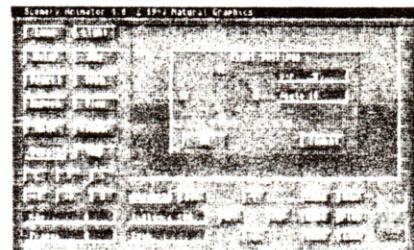
- **SNOW** (hó) metől legyen (mellette lévő érték), vagy legyen e egyáltalán.
- **ROCK** (szikla) szintén metől
- **VEG** (növényzet - zöltség) meddig (ez a vízszintől, illetve 0-tól van)
- **SOIL** (termőtalaj) legyen-é, vagy sem. (ilyen világosbarna foltok)

Megjegyzem, hogy ha nem **HAM8** imagét, vagy **24bit**es **IFF** képet számoltatunk, akkor valamely állítást ki kell kapcsolni, mert különben szól a program, hogy kapcsoljuk ki. (nincs elég szín ??) Vannak még valami **UTM X** és **UTM Y** értékek, ezek növelésével a terep minőségét lehet csökkenteni, 5 körüli értéknél már olyan fasza kockás lesz, mint egy szimulátor. Nem igen piszkáltam őket 0-ról, mert úgy az igazán szép. (igaz hogy sokkal lassabb is). Az ablak alján egy **V. E. FACTOR** felirat figyel. Ezzel a tájképünket tudjuk kidomborítani. (tehát, ha nekünk túl lapos a 1000-es magasság, állítsuk be a **V. E. FACTOR**-t 3-ra, s máris szép hegyeink lesznek. (Ilyenkor a berendezett tájkép - szerencsére - nem változik.)

- Most jön a **SKY** (égbolt, ahol az eget lehet vásárolni). Itt beállíthatjuk, hogy az ég felhős legyen (**CLOUDS**) illetve vegyes (**BLEND**). Ha éjszaka (**NIGHT**) van akkor egyik sem lehet. Itt is be lehet adni egy **SEED** értéket, a felhőzóna magasságát (**ALTITUDE**), illetve sűrűségét (**DENSITY**). Az **EAST** (kelet) és **NORTH** (észak) a felhőzet sebességét jelenti, ellenkező irányba értelemszerűen negatív előjellel kell megadni. (Ugyanis a szél igen ritkán szokott egyszerre fújni északra és délre. Hacsak nem forgószél. Sebaj, majd az 5.0-ban)

- **TREE** : (fák, nem fuck) legyen-é fenyő illetve tölgyfa, - azon kívül, amit felraktunk a térképre - és ha igen, akkor mennyi legyen a **MAX** illetve a **MIN** érték, ami arra vonatkozik, hogy metől meddig nőjenek a fák, (milyen magasságtól, milyen magasságig - default érték 10-3000 - ez egy normális érték, mert ne felejtjük, a fák nem nőnek az égig....) illetőleg mekkora legyen a takarás (takarás, elvtársak), vagyis milyen sűrűn legyenek a fák. (**COVERAGE PERCENT**)

- A **SCREEN**-nél megadhatjuk a **render screen** paramétereit, a színeket is megkeverhetjük (milyen is lehet a vörös tenger - csupa hasznos dolog), meg miegymás. A képernyőméretről már ejtettem szót, én a **640*512**-t használom, de ez mindenkinek az egyéni szoc. problémája. (no meg, hogy mennyi memória van a gépében...) Ha animációt készítünk, érdemes a **FRAME BUFFER**-t a saját **render screen**-ünkhöz passzintani, mert minek egye a memóriát, ha nem szükséges, ugye. A **FRAME BUFFER COLORS**-nál a kettő szín - értelemszerűen - a két határértéket jelzi. A **DCTV DEPTH** a **DCTV ANIM** illetve **PICTURE** rendelésénél kell, ilyen 4 bites faszág beállítása, nem igazán hasznos, mert kurva csúnya képeket lehet vele készíteni. (vagy csak én nem nyesem)



- A **VIEW** segítségével tudjuk megtekinteni a már elkészült művet, ha ilyen nincsen, akkor meglepetésre a **NOTHING TO VIEW** választ kapjuk.

- Az **ANIM** megnyomására elkezd számolni az animációt, (még ilyenkor is megnézi egyszer az egészet, majd szól, ha van valami ütközés, illetve előtte megkérdezi, hogy az összes frame-t ki akarjuk-e számoltatni. Ha **OK** akkor **OK**.) míg a

- **RENDER** lenyomására az éppen beállított (preview ablakban lévő) képet kezdi el renderelni. De még mielőtt mindehhez hozzáfognánk a pull-down menüben állítsuk be az **ANIM MODE**-t (már, ha animációt számolunk. ld.: fent), illetve a **RENDER MODE**-t. A meglepetés annyi, hogy a program tud **PCX** képet is renderelni. (**PCX** animot nem, vajon miért???) Lehet, hogy bé akarnak törni a PC-s piacra. (megnézném ezt a programot egy 14 MHz-s AT-n, hogy milyen sebes lenne.....) A különféle **RENDER MODE**-król csak annyit, hogy az **IFF** képeket a screenen rendereli, míg a **HAM** és a **PCX** képeknél a memóriában számol, és ezt egy **RENDERING IN PROGRESS...** feliratú kis ablakban mutatja, hogy hol tart (**OBJECTS**, **LANDSCAPE**, s végül **SKY**), és az **IFF 24** képeknél a lemezre ír folyamatosan. (A különféle animokat is ilyen módon számolja, csak minden frame-t csak azután ír ki, ha kiszámolta.) Ide igazán tehetek volna egy **TIME ESTIMATE**-t, hogy tudjam, mennyit alhatok, míg kiszámolja az animációt. (Hány napot??)

- Található még itt egy **C** betű, ezzel tudjuk eltörölni az összes frame-t, amit eddig beirgattunk. (szerencsére a program van olyan kultúrált, hogy azért rákérdez) A << és a >> jellel pedig léptethetjük. A + és a - jellel a térképen létrehozott animációkat tehetjük tönkre. Elvileg készíteni is lehet, de minek. Sokkal kevesebb opció van itt. Arról nem is szólva, hogy a térképen némileg egyszerűbb.

- Van még az **OBJECTS** feliratú gomb, ami azt jelöli, hogy legyenek-e objectek a preview screenen. (nem árt ha vannak). A nyilakkal a kamerát mozgathatjuk, illetve a **FRWD**, **BKWD** gombokkal is. És végül a jobb alsó részén a képernyőnek, van kettő kis micsida, amelyek úgy néznek ki, mint egy repülő micsodái. (ott vannak ilyen micsodák a műszerfalon) Ebből a felső az úgynevezett bólintási szög, míg az alsóval állíthatjuk be, hogy milyen ferde is a gerincünk. (milyen ferde szemmel nézünk a világra - hiába a japánok mindenütt ott vannak) Egyébként ezek is beállíthatók a térképnél. A **PLAY** gombbal csinálhatunk egy preview animot is, nem rossz dolog, és amellet elég gyors is. Ha meguntuk, akkor a **PLAY** helyét ideiglenesen elfoglaló **STOP** gombbal leállíthatjuk az egészet.

Még egy kis pull-down menü:

- a **PROJECT** menü nem más, mint egy kis file I/O, de itt a mentés csak a beállított tájképre vonatkozik, magát a renderelt képet a **PICTURE** menüben lehet megőrizni az utókor számára. Elmenteni a tájképet a **SAVE LANDSCAPE**-val érdemes, elmenthetjük azért még a beállításokat (**SAVE SETTINGS**), illetve a palettát (**SAVE PALETTE**). Betölteni a kimentett **DEM** (vigyázat, ez nem dajcsmark - ezt használja pl. a **VISTAPRO** is, többek között) formátumú tájképet lehet, no meg valamilyen **GEO** objectet (meg persze az elmentett beállításokat is), ami betöltés után megjelenik a térkép object listájában. A nyomtatást nem tudtam kipróbálni, mivel nincsen printerem. (amikor ezeket először írtam, akkor nem volt, de most már van, és a kedves felhasználók lelkiüdvéért elmondom: senki ne merjen nyomtatni innen !!! Nagyon csúnya. Meg nem is nyesi. Meg miegymás. Na ezt jól meg mondtam.)

Lehet még összefűzni dolgokat az **APPEND** segítségével, illetőleg lehet a tájképet nyírítani (**CLIP**). A következő kettőről már beszéltem, hátravan még az **OPTIONS**:

- **AVOID** vagy **DETECT COLLOSION**; vizsgálja vagy ne vizsgálja az ütközéseket (az animációnál). Ha vizsgálja, akkor csak annyi, hogy mikor elkezdeném renderelni, akkor visít, hogy van ilyen, és hogy hol, de attól még lehet továbbmenni, csak esetleg meglátjuk a hegyet belülről.

- **CHANGE ALL KEYS** animációnál az összes **KEY** megváltoztatása. (később)

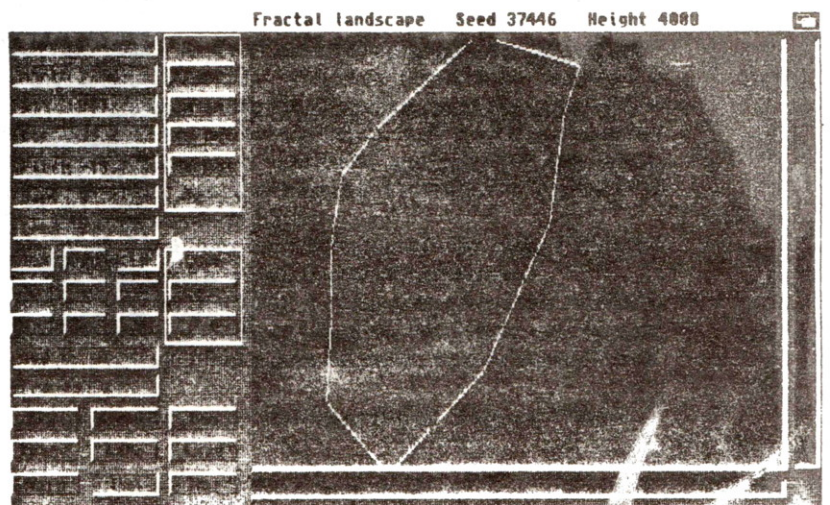
- **AUTO BANKING**, **AUTO BANK %**; kanyarban automatikusan bedőljön-e a kamera, és ha igen, akkor hány százalékban. (default érték az 50) Nem árt átállítani 5-re vagy 10-re, mivel 50-nél igen darabos lesz a bedőlés szöge.

- **LAND BACK FACES**, **OBJECT BACK FACES**; nem árt ezeket bekapcsolni, mert pl.: ha egy import **GEO** objecttel dolgozunk és ki van kapcsolva az **OBJECT BACK FACES**, akkor a kiszámolt objectnek, csak a vázát számolja ki. (nem tömör lesz, hanem vonalából fog állni) De ez a **SCENERY ANIMATOR**-hoz adott **GEO** objectekre nem igaz.

Említettem, hogy lehet objectet importálni máshonnan. Például egy **IMAGINE** objectet.

Be két tölteni hozzá a **VERTEX** c. programot, majd az objectet. Valószínűleg az itt megjelenő object nem igen fog az eredetire hasonlítani, de se baj. (sőt **SOLID MODEL** (gyk.: egy preview az objectről) esetén, - még mindig a **VERTEX 2.0** - sem fog rá hasonlítani, ahogy kinéz. és mégis...) Ekkor el kell menteni az objectet, úgymint **SAVE AS...** és **EXPORT GEO1**. Ez lesz a nyeső. Ezt be lehet tölteni a **SCENERY ANIMATOR**-ba, és lehet renderelni. Egy baj van. Ezek az objectek szürkék lesznek, illetve valószínűleg, ha kirakjuk őket a térképre akkor nyakig a földben lesznek. Ezért majdnem mindig át kell írni az object magasságát. (**FIGYELEM !!** Aki mond egy egyszerű módot, hogy hogyan lehet az import **GEO** objecteket kiszínezni, az kap 5 mezei piros pontot - kitűnően fel lehet őket használni a térképen. De tényleg ! Egyszerű kell, mert bonyolultat én is tudok. Amellet bármilyen objectet szívesen látok. Nekem is van egy pár fölös.)

S végül, amire már mindenki várt, itt van az **ANIMÁCIÓ**; Menjünk a **MAP**-hez. Itt legelőször be kell állítani, hogy két **KEY** között hány **FRAME** legyen (ha nem tesszük az elején, csak mikor már elkezdjük csinálni az animációt, a program **YOU NEED AT LEAST ONE FRAME** - legalább 1 frame köllene - felirattal örvendeztet meg minket, mivel a 0-s **KEY**-nél nem lesz **FRAME NUMBER** - ez egy nagyon gecsi buktatója a programnak, mert első alkalommal én is itt nyomtam **RESET**-et, mert nem tudtam, hogy mi a fasz baja van...), magyarul, kettő, általunk beállított kamera pont között hány **FRAME**-t számoljon ki a gép. (ez attól is függ, mekkora távolságban vannak egymástól a pontok). Ezután le kell tenni a kamerát a térképre. Meg kell nyomni a + gombot, majd arrébb kell tenni, és megint megnyomni a + gombot. Ez csak leírva hangzik ilyen faszul, csinálni tök egyszerű. Ha az **LP (LOOP)** be van kapcsolva, azonnal hurkot csinál, de ez senki se zavarjon meg.

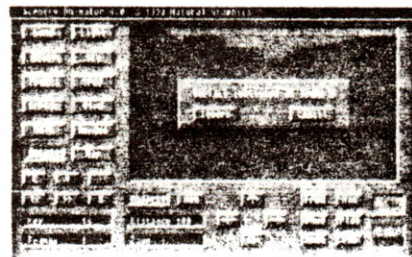


Amennyiben a **CURVES** is be van kapcsolva (legyen bekapcsolva), akkor a program kanyarokká számolja a kockafordulásokat. (ami szépség szempontjából nem egy hülyeség). A **PRO** gombbal megnézhetjük, hogyan is fest oldalról az animáció. (már a szintkülönbségek, meg miegyebek) Vissza a térképre a **MAP**-pal mehetünk, mert az az igazi. (főleg, ha jéghideg.)

Mikor mindezekkel megvolnánk, és úgy érezzük, van annyi hely még a winchesterünkön, amekkora az animáció lesz, akkor az **XTEND** gomb lenyomásával véglesítheljük a dolgot. Ekkor a gép a **KEY**-ek közé behelyezi a megadott számú **FRAME**-kat. Természetesen itt is le tudjuk játszani az animációt a **>>** és a **<<** nyilakkal. (A főmenüben létrehozott animáció készítési metódusa is hasonló, mondhatni ugyanez. Vagyis beállítjuk a **FRAME**-k számát, majd a nyilakkal, illetve a **FRWD**, **BKWD** (nem a BKV d) gombok, és a **+** gomb lenyomásával gyárthatjuk az animot. Itt sajnos nem lehet olyan fasza kis kanyarokat venni, de legalább látunk valamit. Ha végeztünk, vissza kék menni a térképhez egy **XTEND** erejéig. Szerintem sok teteje nincsen a dolog illetén való használatának.)

És mehetünk is vissza a **MAIN**-ba. Ha valami karambol van az animációban, (és be van állítva a **DETECT COLLOSION**) akkor a program kukorékol, ilyenkor a preview ablakban nem árt megnézni a dolgot, (main menu-ben **PLAY**) mivel nem mindig olyan nagy a baj. Eztán nincs más dolgunk, mint elindítani az **ANIM** gombbal az animáció számolását. (addig el is mehetünk valahová.)

1200-esen (FPU nélkül 4 MB FastRammal) egy 30 frame-ből álló animáció 320*256-on IFF ANIM módban olyan fél óra - 40 perc körüli idő. (még nem mértem) De hát objectek, meg miegyebek nélkül. No meg a felbontás sem az igazi. Az elkészült file hossza olyan 700K körüli lesz. (de ezt pontosan meg tudom majd mondani, mert éppen most számol egyet a gépem) Pont 601K hosszú lett (320*256, IFF ANIM, UTM = 5, no objects), de ez felbontásbeli javítások esetén, valamint behelyezvén 100-200 objectet (minek aprózzuk el), ugrásszerűen meg fog nőni.



Mindezeket összevetve a **NATURAL GRAPHIC 1993**-ban kiadott programja nem olyan rossz. Az FPU-s verziót még nem próbáltam, de remélem nemsokára lesz hozzá cseresznyém. A programmal kettő baj van talán igazán.

- 1.- Igen nehezen kezelhető elsőre. Meg másodikra, meg harmadikra... (nem egy felhasználóbarát) Emellett igen csúnyácska is szegény (csaknem Windows like - na nem, azért annyira mégsem)
- 2.- Egy valós képet még **HAM8**-ban sem igen lehet létrehozni. (csak igen hasonlót hozzá) Talán, ha volna több object....

Rosszúságok: Nem lehet lelőni a számolást, amíg az objecteket számolja.

A térképen nincsen **UNDO**. (néha igencsak hiányzik)

Eléggye jól meg van csinálva az animációkészítő része. (ez azért baj, mert csaknem kezelhetetlen. No de azért, ha az ember belejön már könnyebb. De az elején én is rengeteget szoptam.)

Hiretelen ennyi, de nem kell megijedni, fog mindenkit érni még egy jópár úgy közben is.

Az animáció méretéről már beszéltem, vagyishogy részben; egy **HAM8**-ban renderelt kép olyan 200 körüli objecttel, 640*512-es felbontásnál jó 300K, átszámolva **JPEG**-be 150K körüli méretet fogunk kapni. Én talán hiányoltam belőle a **JPEG** savert, igazán beletehettek volna a **PCX** helyett. (Igaz így lehet, hogy elvesztette volna nagyfokú Windows kompatibilitását, ami biztosan komoly problémákat okozott volna mindenkinek a lelkivilágában.)

Ahhoz képest, hogy milyen "komolytalan", nem is olyan gagyi program ez. (még mindig csinálhatunk vele olyan animot, amelytől leesik minden családtagunk álla - csak győzzük kivárni) Igaz a **VISTA** szebb is meg jobb is, de az igencsak "FPU igényes" dolog, no meg kissé kevés neki az **MC68020** is, sőt mitöbb. illik hozzá úgy 8 Mbyte FastMem is. (Ha nem több, mert 4 Mbyte-n eléggé...izé.) De egy "fapados" 1200-en ez is jó. Gyártottam már 32 színű IFF animációt vele **LOWRES**-ben (320*256) és úgy összességében azért jól nézett ki. Nyesni kell... Meg szeretni is. Van egy 4.1-em is, de az is csak 881-es verzió... Ha fogok rendelkezni ilyennel, akkor majd leirom, hogy mitől jobb. Ha meg lesz valami "erőgépm" akkor írok majd a **VISTA PRO**-ról is....

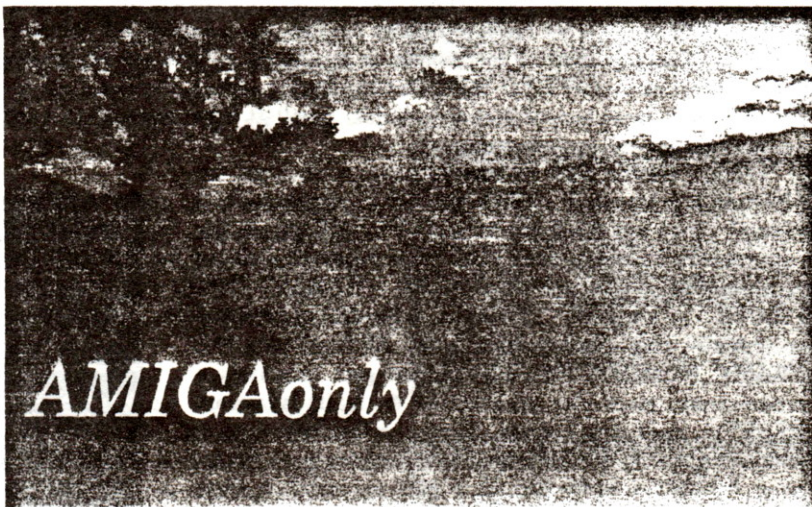
Itt volt vége az eredeti leírásnak, de azóta eltelt egy-két hónap. (kb. kettő) Magában a leírásban - biztosan - lesz néhány hiányosság, de ha ezeket bekülditek - levél formájában a postafiókunkba - akkor esetleg folytatólagosan le is közöljük azokat, hadd okuljon a jónép.

Magic

(e heti renderelt képünk :

"tehénke a hegyoldalban, délben"

Mindenkinek minden jót!)



How 2 Use a Fanatic

avagy mit kezdünk egy megszállott AMIGA tulajdonossal

Úgy is kezdek az egészet, hogy: Együttelési tanácsadó a számítógéppel és a pároddal. Azért nem iriam nemet, mert legyen az illető parod -nő, illetve férfi, a szisztema mindig ugyanaz. Lehet, hogy sokakban felmerül a kérdés: - Miért kell egyszerűen tanácsadás ehhez? Hisz csak egy gép!

Nos ezek az emberek, meg ha nem is tudnak róla, hatalmasat tevednek. A számítógép nem csak egy gép - hanem vetélytárs, ami elveszi tőled a partnered, mivel minden idejét vele tölti, s ha egy csekélyke időt szentel rád, akkor is csak róla beszél, Őt magasztalja, Őt dicsőíti, s csak körülötte forog a világ. Tehát te magadat - egy harmadik feleslegesnek tartod. S ekkor már nem gép - hanem olyan dolog, ami megkeseríti ele - ted - ha hagyod.

Sok válasz, tragedia történt már a párkapcsolatokban a gép miatt. De ha nem vetélytársnak, ellenségnek tekintjük a számítógépet, akkor úgy kell gondolkodnunk - mint Ő.

Gondolom, hogy most felteszitek magatoknak azt a kérdést - "Hisz Ő nem tud gondolkodni!?". Gyakorlatban nem - de lelke van - s olyat tud - amit az embertől kapott - tehát felruháztuk Őt emberi tulajdonságokkal, s ha belegondolunk - teljesen behalózza a társunkat, beleköltözik az agytekevényeiben, lassan, óvatosan, s Ő, a nagy Ő ebből semmit sem vesz észre. Csak Őt látja, Ő létezik számára, s tudata csak Ő körülötte forog, s nem érti, hogy te nem érted ezt csodálatos érzést. Ő nem olyan, mint egy emberi lény, sokkal többet tud - mondja a párod. Olyan teremtmény ő ami csak neked el, teljesíti a kívánságod, feleslegesen nem kérdezősködik, egyértelműen fejezi ki magát, nem kéri számon hibáidat, nemszámoltat be, hogy hol jártál, mit csináltál, hol s mivel töltötted az éjszakáidat, csöndben van, nem felesel vissza, nem csinál jeleneteket, csak kiszolgál, teljesíti parancsaidat minden hiba és kérdés nélkül. Hát igen! - hol találunk még egy ilyen tökéletes párt? Es nekünk ezzel a tökéjjel kell megosztanunk az életünk, s mindehhez mosolyognunk kell.

De nem muszály! Hisz mi is megszerethetjük, mi is fölthetünk kellemes estéket, velünk is éppúgy szeretettel bánik, mint a párunkkal. Csak érteni kell hozzá. Felesleges negatív gondolatokkal taplálni teleje - hisz gép - de lelke van. Mi tettük az ember szolgáljává, hogy mi emberek szolgáljuk Őt. Mi nem létezhetnének már az Ő tudásuk nélkül, s Ők se létezhetnének a mi tudásunk nélkül.

Ez egy ördögi kortorgás - amit mi teremtünk olyanra, milyen az ember szeretett volna lenni ő tökéletes. De ne feledjük, - Ő sem tökéletes - csak velünk együtt.

Ha félsz tőle, vagy haragszol rá, vagy éppen a féltékenység györöt, vagy gyűlölöd, s remélem azért akad olyan társ, aki szereti, s olyan ember is, aki semleges vele szemben - hisz gép - azért kerlek benneteket próbáljátok ki - mikor kettesben vagytok - milyen együtt a számítógéppel.

Leültök vele szembe, s a mélyen a szemébe nezték (ebben az esetben a monitor) s molyogjatok rá, hisz a bizalom a legfontosabb. Ő megérzi, hogy békével közeledsz hozzá.

Kapcsoljátok be a gépet, a monitort - s adjátok át magatokat az Ő ébredésének, s lágyan simogassátok meg Őt. Érezni fogjátok szívetekkel a benne futó áramköröket, ami olyan, mint a bennünk folyó vér. Életet adtunk neki, tehát el is vehetjük. Nincs mitől félni. Beszélgethetünk vele, játszhatunk vele, utasításokat adhatunk neki, rajzolhatunk vele, logikai feladatokat oldhatunk meg vele, s bármilyen bonyolult matematikai képleteket oldhatunk meg a segítségével. A társunk lesz addig, amíg mi akarjuk. Együtt leszünk urak és szolgálók. Jól fogjuk magunkat érezni vele. S ha úgy érezzük, megtaláltuk mi is azt, amit a társunk - nem leszünk rá féltékenyek, mert nekünk is azt nyújtotta, amit neki.

S ne feledjük, közömbösek se lehetünk feléje, mert ha nem hallgatjuk meg párunkat a vele történt izgalmakról, meg hogy együtt milyen problémákat oldottak meg, előbb vagy utóbb egy olyan embert választ, akivel mindez megoszthatja, s akkor ott fogunk állni, s még csalódottabbak leszünk, még kirekesztettebbek, s még jobban fogjuk gyűlölni a számítógépet - pedig nem Ő a hibás - hanem mi - mert valakinek engedni kell, s hogy miért pont nekünk? Mert olyan kontaktus alakulhat ki köztünk, ami csodálatos is lehet, meg hasznos is. Mert ha mi földi halandók belegondolunk, hogy mennyi előnnyel jár ez - azt nem is gondolnánk. Mert tulajdonképpen tudjuk mindig (legalább is általában), hogy a társunk hol van és mit csinál. Igen - a számítógép előtt ül.

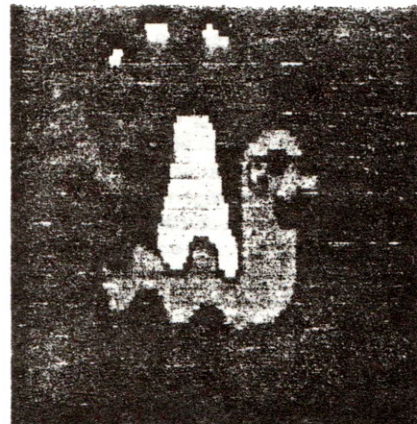
Legyen az éjszaka, hajnali három, vagy délután négy. Tehát a hely és az időpont adott. Azt is tudjuk mindig, hogy mit csinál. És ezzel már előnyünk van a többi emberrel szemben, akik általában sohasem látják a párjukat, s nem tudják, hogy mivel töltik az idejüket, s kivel.

Mert mi mindig az asztal mellett, a számítógép társaságában találjuk őket. S a negyedik személytől nemigen kell félnünk, mert a gép meg mi - ha ügyesek vagyunk - felemészítjük az összes energiáit. Tehát csaki mi ketten "birtokolhatjuk" ugyanazt a személyt. Mi tehát tudjuk, ismerjük a vetélytársunkat, s így könnyű dolgunk van. Mert ha belegondolunk, nincs semmi veszéllyel ránk. Mert amit egy számítógép nyújt, s adni tud - azt mi nem adhatjuk meg - de - ez a legfontosabb, hogy **amit a gép nem tud nyújtani - mi azt mind megadhatjuk neki**. Így kiegészítjük egymást. Mind azt a szeretetet, gyengédséget, simogatást, jókedvet, nevelést, az emberi élet minden örömet és bánatát. Mert nekünk nem kell bonyolult számításokat, egyenleteket megoldanunk, nem utasításokkal működünk. Mi szabadok vagyunk, de Ő az áramkörök rabja.

Tehát nem kell haragodnunk a számítógépre - mert mind a ketten mást nyújtunk - másféle problémákat oldunk meg, s másképp közeledünk a partnerünkhöz. Hisz **mi is olyan értéket képezünk párunk szemébe, s nem vagyunk kevesebbek**.

Meg kell őket érteni, mert ők mások. Van aki az alkohol rabja, van aki az erőszak rabja, a pénz, a női nem rabja. Mindenki valami rabja. Ők a számítógép rabjai, csendes örültek, akik szeretnek elbújni a gépükkel egy csendes sötét zugba, csak hagyják őket békén. S ezért nagyon hálásak tudnak lenni, s meglátjátok, hogy örömmel térnek vissza hozzátok, ha nem bántjátok őket. Olyanok, mint a kisgyermek. Játsszanak.

Mi is tanuljunk meg velük játszani, hisz ezzel nem bántanak meg senkit, s örülni fognak, ha leülsz melléjük, s figyelni fogod a



kettejük örömét, amiben te is egyenrangú partnerként részt vehetsz. S lassan rá fogsz jönni, hogy - lehet együttelni a párod számítógépével is.

Néhány Útmutatás az együttélés megkönnyítéséhez, illetve annak megtartásához;

- Megismerkedésekkor, mikor tudomást szerzel arról a tényről, hogy számítógéppel dolgozik, vagy otthon van neki - lehetőleg ne vágj savanyú képet, mint aki ecetes uborkába harapott, vagy ne kezdj el örülni nevetni, mert ezzel elriaszthatod magad mellől, mielőtt megismerhetted volna.
- Soha ne szidd a számítógépet - mert az nekik "Szent és sérthetetlen", mert azzal magad alatt vágod a fát, és ezzel nem fogsz a szívéhez közel kerülni.
- Soha ne is dicséred agyon a számítógépet, mert pár szóból egyből rá fog jönni, hogy nem is láttál előtte számítógépet soha.
- Ha hét napból csak háromt akar veled tölteni, soha ne vágd a fejéhez, - hogy: persze, mert a géped fontosabb neked, mint én! Ettől nem fog többet találkozni veled, s lehet, hogy a fennmaradó hármát is megfogja gondolni, s ezt te nem így akarod, ugye?
- Ha beszélgetek, nyugodtan hallgasd végig a géppel kapcsolatos élményeit, mert elsősorban tanulhaszt belőlük, s ha egy ilyen társaságban találod magad, legalább sejtelméd lesz, hogy miről beszélgetnek.
- Lehetőleg ne vágj fancsali képet, ha unatkozol, inkább kérd meg őket, hogy avassanak bele téged is, vagy beszéljenek veled érthetőbb dolgokról, de semmi esetre ne csinálj hisztérikus jelenetet. Az olyan kínos.
- Ha két, vagy több számítógépes zseni összejön pl. nálatok, nyugodtan ott hagyhatod őket, s elmehetsz szépen moziba, s megnézheted azt a filmet, amire már régen vágytál.
- Ha késő este hazamész, s párod ott gubbaszt a számítógép előtt, mint "varjú a fán", sohasz szidd le érte. Gondold úgy, hogy téged várt eddig, csak lefoglalta magát valamivel.
- Ha valami problémája van a párodnak egy megoldatlan ügyben, amire nem talál megoldást, hallgasd végig, mert egy laikus, mint kívülálló, sokszor teljesen más módon gondolkodik, s hamarabb megtalálja a megoldást, mint Ő, mert hidd el, Ők sokkal bonyolultabban gondolkodnak, s ez rájuk nézve néha veszélyes. De mi tudunk nekik segíteni.

Még sorolhatnánk olyan gondolatokat, amelyek megkönnyítik az együttélést, de szeretném, ha ti is kifejtétek a gondolataitokat, mondanivalókat, hogy hogyan lehetne ezt a problémát, még ha megoldani nem is, de megkönnyíteni, hogy lehetséges. Ha valakinek van valami elmés megoldása, ami jól bevált és kivetelezhető, továbbá másokkal is szívesen is megosztaná, kérem az írjon az újság címére.

Ha hozzájárulsz ahhoz, hogy ötleteidet leközzöjűk, akkor köszönettel vesszük mi - akiknek együtt kell élni a párnk számítógépével, annak minden örömeivel, szomorúságával, előnyeivel és hátrányaival.

Ami fontos: Soha ne állítsuk őket válaszüte elé: "Vagy Ő, vagy Én", mert lehet, hogy az "Ő" mellett döntenek, s ez nagyon tájé pont lenne nekünk. Így meg van a lehetőségünk, hogy mi is fontosabban lehetünk, csak okosan kell a dolgokhoz állnunk.

Bár lehet, hogy az "AMIGA szent és sérthetetlen", de ne feledjük, hogy mi viszont emberek vagyunk, és Ő egy ember alkotta gép, tehát nem (Ator) a legyőzhetetlen.

Üdvözlettel: Lya

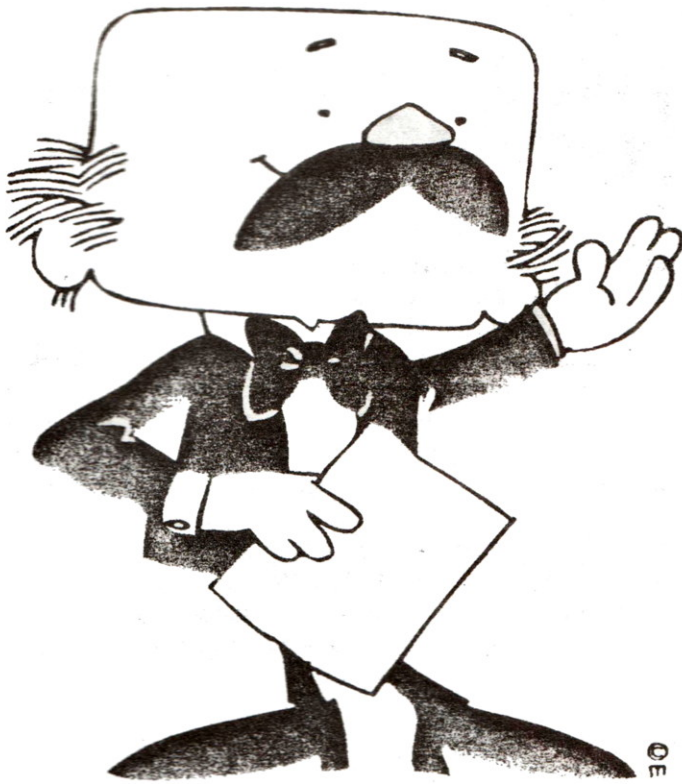
Szerkesztőségünk keresi a következő szabadon terjeszhető software-keket:
(mivel szeretnénk a "rég elfeledett" programokat egy CD-re feltenni)

- régi (500-as ra íródott kaland-játékokat(1,2,3,4,5,10 lemezesekeket) KQ, SQ, Larry etc...

- Future Composer zeneszerkesztőt (lehetőleg olyat ami bír működni A1200-esen...)

- vállalkozókedvű amatőrt, aki hajlandó a PC-Task-kal betölteni a Windows alatt futó valamely extravagáns progit, aztán lefagyasztani és screengrab-olni...





SQUIRRL-PCMCIA-SCSI- 2-HOST-ADAPTER

Mint ahogy a cím is elárulja, egy különleges SCSI-csatlakozó jelent meg az AMIGA 600 és 1200 PCMCIA-SLOT-jára. Ehhez tudni kell azt, hogy a PCMCIA-nak csak 16 bites a busza, ezzel ellentétben az A1200 32 bites, tehát a géphez képest ez a port nagyon lassú. Ennek ellenére kellemes csalódás érheti a vásárlót ha leperkálja a 215DM-et az AMIGAoberland-nak. Ugyanis egy alapképzésű A1200-on 1,5 MBYTE/SEC sebességet képes produkálni, ami igen szép teljesítménynek számít.

Akinek valamilyen turbókártya van a tulajdonában az élvezheti a 3MBYTE/SEC-os átviteli sebesség előnyeit.

Az SCSI-portnak igen sok előnye van, többek között a sebessége és a nagy flexibilitása, hiszen nem csak winchester-t hanem streamer-t, DAT-ot, scanner-t és mindenképp előtt CD-ROM-ot is tudunk vele használni.

Erre a fejlesztők is gondoltak és ezért bőséges szoftverrel látták el a csomagot, amiből a CD32-emulátort emelném ki, ami 90%-os kompatibilitással dolgozik.

A PCMCIA-dugó kb. telefonkártya nagyságú, és könnyen illeszthető. A kártya kilógó része 4 kis láboscán támaszkodik az asztalra, ami a stabilitást biztosítja. Innen egy 50pólusú centronics kábel vezet az SCSI egységhez.

Összegezve nagyon jó benyomást kelt ez a kutyú, az egyedüli problémát talán csak a helyigénye és zene CD lejátszó program primitívsége okozhatja. Ezeket a problémákat egy Eagle-towerrel és egy PD lejátszóval segíthetünk.

Megrendelő:

AMIGAoberland
61476
Kronberg Schneihohl 5(PF)
Tel: 0-61-73-65001

HD-SpeedUP

Mint tudva-levő a régi Commodore cégnek a spórolás volt a mániája, és ennek már igen sok bajt láttak az AMIGA felhasználók. Egyike ezeknek az "áldásoknak" a Winchester-vezérlő, amely nem túl gyors, és még a régi szabvány szerint működik. Erre ad segítséget a német VOB cég SpeedUp nevű terméke, amely talán mankóként szolgálhat az AMIGA tulajdonosoknak.

Ez a Hardwer lehetővé teszi a HD minimum 50%-os gyorsulását, és egyszerre 4!!! AT-buszos egység használatát. A Winchester gyorsulása függ a típusától, és a használt gép kiépítésétől, ezáltal akár 7-szeres gyorsulás is elérhető!!! Ezekon kívül még az új ATAPI (AT Advanced Packed Interface) szabványt is támogatja.

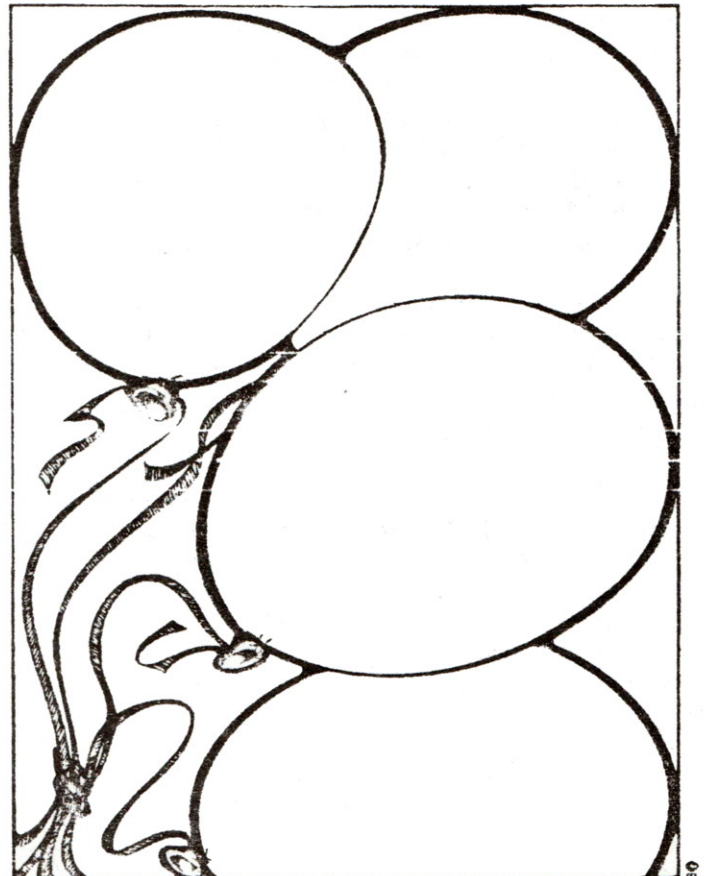
A csomaghoz tartozik egy install-lemez, a HD elosztó és egy AT/IDE kábel. Installálás során a program átírja a Startup-sequence-et, hogy mindig elinduljon a meghajtópatch ami lehetővé teszi a 4 HD kezelését, ezért ezt soha ne írjuk át mert gondok akadhatnak. Egyébként semmi különösét nem kell vele csinálni, észre sem veszi az ember, olyan, mint ha csak több partíció lenne. Az adaptert egy kétoldalú szigetelőszalaggal (hozzáadják) kell a táphoz erősíteni. Ezen van egy kábel aminek két végén van egy-egy fűsor amikre egy kétféle osztott kábelt lehet tenni (egyét adnak hozzá).

Aki mindezek után nem rohan 98DM-el Németországba, hogy megvegye, azt csak sajnálni tudom, mert ilyen lehetőséget nem szabad kihagyni.

Megrendelhető:

VOB Computersysteme GMBH
Postfach 1000607
44006 Dortmund
TEL: (02 31) 9-12-21-46

FRANKISS



Intel Ouside - Az igazak harca

Mottó: A PC csak addig tűnik jónak amíg nincs! Utána már úgys mindegy.

Amióta csak számítógépek vannak, a cégek mindig is megpóálták gépeiket ráerősokolni a vásárlóközönségre. A nagy géparadot elvakította az embereket, nem tudták, hogy melyiket vegyék, ezért az IBM cég egy olyan moduláris gépet létrehozni amelyik nem avul el mert részei cserélhetőek. Csak egyre nem gondolt:

az új moduloknak, perifériáknak is kompatibilisnak kell lenniük a régiekkel, de nem csak a pozitívumokkal hanem minden negatív tényezővel is. Ez nagyon gátolja a fejlődést - meg is látszik a 20 év, ennyi idő alatt jóval többet fejlődött a technika, mint ahogy a PC-k. Tehát még ma - 95-ben - is a 70-es években kitalált szisztéma szerint építik a gépeket. Hogy érthetőbb legyen mondanék pár példát:

- A 70-es évek elején még nem gondoltak a jövő memóriagényére, ezért úgy gondolták, hogy 1MB bőven megteszi maximális címzésnek, csak hogy, mint tudjuk azóta ez változott és ma már a 4MB sem elég egy játékhoz, ezért mindenféle memóriamanagerrel kell bajlódni, hogy végre elinduljon a kívánt program.

- Ez még hagyján, de az hogy azt az 1MB-ot 64KB-ként kell lapozgatni, és el lehet képzelni, hogy mekkora sebességet és kezelhetőséget tud produkálni.

Ezzel szemben az AMIGA 128MB-ig autoconfigos és nem kell lapozni. Egy A4000-be 1808MB memóriát tehetünk!!!! (7*Zorro-n 256+64 az alapon)

- Ugyanúgy nem gondoltak a háttérterek növekvő megabájjtaira sem, ezért ma minden 540MB fölötti Winchesterhez adnak egy install-lemezt amellyel talán felismeri a gép a HD többi részét, azért csak 'talán' mert ez az alaplaptól függ.

- Mivel a gép nem játékra készült, ezért természetesen semmiféle képkezelést nem tud, minden csak attól függ, hogy a programozó ki tudja-e cselezni a procit valamilyen matematikai trükkkel. Ilyen például az X-MODE nevű rutin amely még így is nevetségesen scrolloz akármilyen gyors is a gép amin fut.

- Ha valaki egy régebbi gép tulajdonosa akkor aztán keresheti rajta a joyportot tudnillik azt is meg kell venni (az új I/O kártyákon már van egy). Mondjuk valaki vett magának egy joy-t, egyszerűen csak feljön a haver, hogy szeretne Mortal-ozni. Csak hogy hova dugja ekkor a második joy-t, hiszen 2 I/O kártyát nem használhat egyszerre, erre is van megoldás: egy ismeretlen cég, ismeretlen hangkártyája amit csak külföldön lehet beszerezni, de ugyebár azon van még egy port. Na igen, de mi van akkor, ha a program nem támogatja a második joyt ?!

És akkor még nem is beszéltünk a joy használhatatlanságáról. Mindezek miatt egy PC-s sem használ szívesen joy-t, aki nem hiszi járjon utána.

- A programozók dolgát még az is nehezíti, hogy nagyon sok hangkártya terjedt el az évek során, és ha nem akarnak csődöt akkor lehetőleg minnél több hangkártya típust ismerjen és használjon a programja. Emiatt írdatlan hosszú lehet egy ilyen proggy. Grafikánál defto.

Na jó, ezt még a végtelenségig is lehetne folytatni, ezért más szempontból is megnézzük. A PC-t sokan Multimédiás gépnek hiszik, de csak azok, akik nem tudják, hogy mit jelent, mert szerintük az, amikor van hang is, meg kép is, és egy CD-rom. Ezzel az elvel még a C64 is multimédiás gépnek

számít (ahoz is van CD-rom, meg van hang és kép). A Multimédia valójában az amikor egy gépet nem csak játékra meg szövegszerkesztésre lehet használni, hanem például egy videófilm vágásához, szerkesztéséhez, professzionális zenefeldolgozásához, újságszerkesztéshez élethű animációk egyszerű elkészítéséhez. Ha ezeket egy gép együtt tudja akkor beszélhetünk MULTImédiás (MULTI=több) gépről. Ilyen gép például az AMIGA és a MACintosh, de semmikép sem a PC-k. Olyan gépet pedig főképp nem amelyik 95-ben 320 x 200-as képernyőn dolgozik. Az ilyen, és ehhez hasonló gondolkodású emberek sem tudják felfogni, hogy nem a proceszor sebessége a domináló mindenben, mert akármilyen gyors is, egyszerre csak egy dolgot tud csinálni, ezért nincs PC-n valódi Multitasking.

A ZX Spectrum is hasonló elveken működött, ezért bukott meg. A C64 pedig még ma is egész jól tartja magát, ellenben a spectrumok már 5-6 éve eltűntek. Lehet asszociálni és reméljük így is lesz, mert az arany szabály mindig is ugyanaz marad:

Egy proceszor csupaszon semmit sem ér.

Ne felejtsetek az AMIGÁK még messze nincsenek kihasználva, csak hát a cégek programjaikat úgy adják ki, hogy lemezről is fussanak nem pedig HD-ről, ez az egyetlen nagy hátrány a PC-vel szemben és ezen csak ti a felhasználók és játékosok segíthettek ha, legalább egy 120MB-os Winchester-t csatolnátok gépetekhez és emiatt nem kéne másoknak szenvedniük. Higgyétek el az Amiga sokkal többet érdemel.

FRANKISS

móltör

Ez az újság nem jöhetett volna létre, ha néztünk volna a Frici Sót, meg a Pallaszt!

Igen Nagy üdvözléteket küldjük az ország ige Sok pontjára, minden hasonszôrű kollégának...

Például:

*Louise / AMIGAonly
Magic / AMIGAonly
Lya / AMIGAonly
& FRANKISS & -mAsH-*