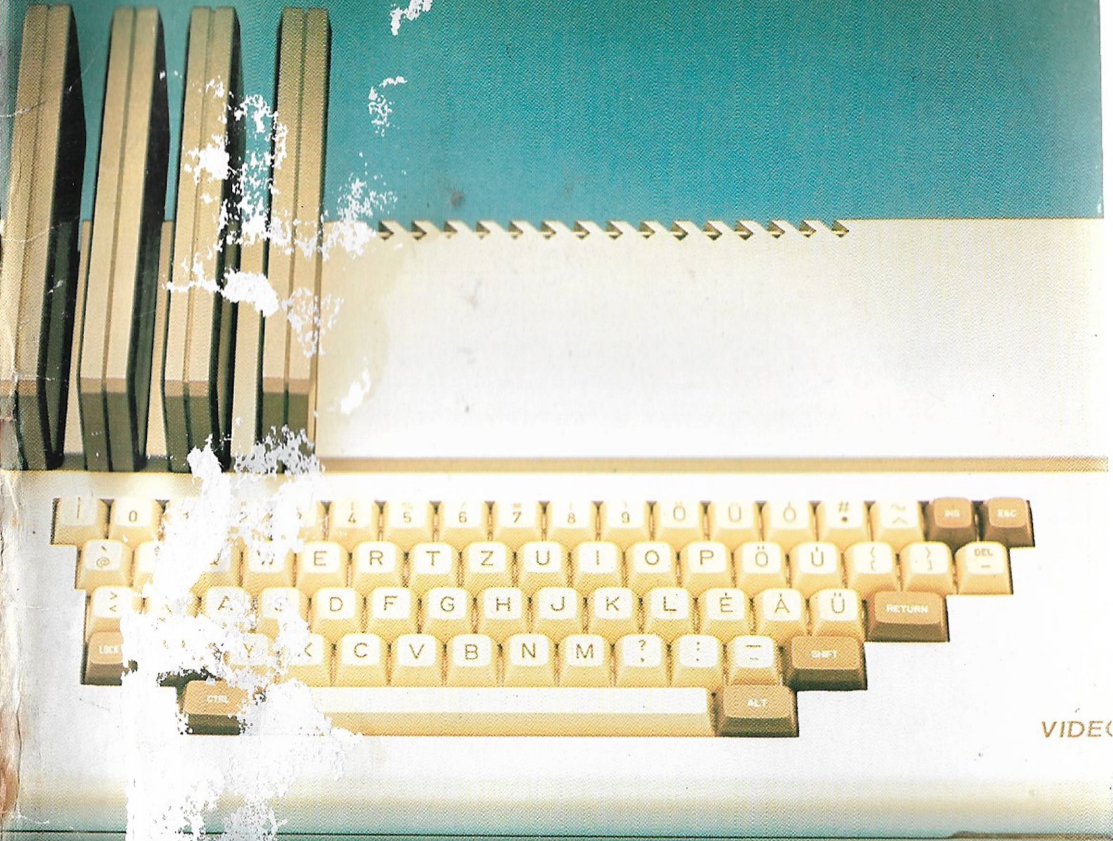


MEM  
adatkezelő program



VI  
TON

T  
Computer



**TV COMPUTER**

**M E M O**

**ADATKEZELŐ RENDSZER**

**FELHASZNÁLÓI KÉZIKÖNYV**



Minden jog fenntartva!

Felelős kiadó: Dr. BARÁTH CSABA

Készült: Forma-Art



## TARTALOM

Bevezetés . . . . .	5
Mi a MEMO? . . . . .	5
Mi kell a MEMO használatához? . . . . .	5
Mire használható a MEMO? . . . . .	6
Ki tudja használni a MEMO-t? . . . . .	6
Kedves Felhasználó! . . . . .	7
A leírásban használt fogalmak magyarázata . . . . .	9
A MEMO parancsai . . . . .	15
Parancs menü . . . . .	16
Adatbevitel-módosítás . . . . .	17
Kijelölés . . . . .	20
Törlés . . . . .	23
Összegzés . . . . .	25
Listázás . . . . .	26
Nyomtatás . . . . .	27
Sor kiválasztás . . . . .	28
Rendezés . . . . .	29
Egyéb parancsok . . . . .	30
Hogyan használjuk a MEMO programot? . . . . .	34
Általános tudnivalók . . . . .	34
A program indítása . . . . .	35
Példa a MEMO használatára . . . . .	36
A MEMO korlátai . . . . .	40
Különleges helyzetek lekezelése . . . . .	40



## BEVEZETÉS

### Mi a MEMO?

A MEMO egyszerűen kezelhető, maximálisan rugalmas program. Segítségével tetszőleges – de egy adattáron belül azonos – szerkezetű rekordokból álló adathalmazt építhetünk fel és kezelhetünk.

A MEMO működése közben az adatokat a számítógép memóriájában tartja, s ezzel biztosítja, hogy a program kezelője még viszonylag nagy mennyiségű adat esetén is azonnali válaszokat kapjon. A memóriában tárolt adatokat kiírhatjuk kazettára. Egy kazettán több, különböző szerkezetű „MEMO” adattár is tárolható. A kazettán tárolt adattárakat később újra beolvashatjuk a számítógép memóriájába, ahol a régi adatokat módosíthatjuk, új adatokkal egészíthetjük ki memónkat.

A tárolt adatokból gyorsan és egyszerűen választ kaphatunk kérdéseinkre: kikereshetünk adott tulajdonságú adatokat, sorba rendezhetjük adatainkat a magyar abc ill. számok növekvő sorrendje szerint, kérhetjük a számadatok összegzését, stb.

A kiválogatott adatokat kiírhatjuk a televízió képernyőjére, vagy nyomtatóra.

### Mi kell a MEMO használatához?

A MEMO program használatához nem kell más, mint egy TV Computer, egy televízió és egy kazettás magnó.

A program úgy lett megtervezve és kidolgozva, hogy optimálisan kihasználja a TV Computer memóriáját.

Mivel a program a számítógép memóriájában tárolja az adatokat feldolgozás közben, ezért természetesen több adat egyidejű kezelése lehetséges 64 K RAM-mal rendelkező gépeken, mint 32 K RAM memória esetén. Az utóbbi esetben is nagyon sok feladatra használhatjuk programunkat. A MEMO korlátai c. fejezet mutatja be a lehetőségeket.

Amennyiben nyomtatóval is rendelkezünk, a tárolt adatokat és az eredményeket nemcsak a képernyőn jeleníthetjük meg, hanem ki is nyomtathatjuk.



## Mire használható a MEMO?

A MEMO segítségével kisebb-közepes adatnyilvántartási feladatok oldhatók meg.

Néhány tipikus alkalmazási terület:

- telefonjegyzék
- ügyféljegyzék
- fizetési jegyzék
- kiadványok jegyzéke
- személyi adatok nyilvántartása
- havi kiadások nyilvántartása

A felsorolásból látható, hogy a MEMO hatékony segítség számos feladatnál mind otthon, mind a munkahelyeken.

## Ki tudja használni a MEMO-t?

A MEMO program kezelését bárki elsajátíthatja minden előzetes számítástechnikai ismeret nélkül is.

Azok számára is ajánljuk a MEMO programot, akik eddig csak kész játékprogramok futtatására használták számítógépüket.

Ez lehet a számítógéppel való ismerkedés következő lépcsője.

A program megismeréséhez javasoljuk először átolvasni ezen leírást, melyet igyekeztünk úgy megfogalmazni és sok példával kiegészíteni, hogy ne csak programozók, hanem a számítógéppel még csak ismerkedők számára is érthető legyen. Ezért pl. külön fejezetben megmagyarázzuk a leírásban használt fogalmakat.

A program használatát a program maga is megkönnyíti. A program irányítása úgy történik, hogy magyar nyelven feltett kérdésekre kell válaszolni ill. a kívánt funkciót kifejező magyar szó első betűit kell begépelnünk. A párbeszédés üzemmód segítségével néhány kisebb feladat végrehajtása után már a leírás nélkül is fogja tudni kezelni a programot.

Ha mégis elfelejtené valamelyik funkció kódját, jön a segítség. A "segít" szó begépelésére kiíródik az összes funkció (parancs) kód.

Az ön által vásárolt kazetta a programon kívül tartalmaz egy "könyvek" nevű



példa adattárat is. Javasoljuk, hogy a program kipróbálásakor hajtsa végre a "Példa a MEMO használatára" c. fejezetben megadott lépéseket, mely végigvezeti önt a legfontosabb funkciókon és segít megérteni a funkciók közötti kapcsolatokat.

További gyakorlás céljából érdemes kipróbálni a parancsoknál közölt példákat.

### **Kedves Felhasználó!**

Reméljük, sikerült felkelteni érdeklődését és már alig várja, hogy kipróbálhassa a programot. Kívánjuk, hogy sok feladat megoldásában legyen önnek segítségére a MEMO.

Fel szeretnénk hívni figyelmét a programmal kapcsolatos jogaira. A TV Computerre kidolgozott MEMO program a VIDEOTON tulajdona. A programkazetta és leírás megvásárlásával ön jogot szerzett arra, hogy azt TV Computerén használja. A program azonban nem másolható le, nem adható tovább! Ugyancsak tilos a program vagy annak részeinek átírása más számítógépre a program szerzőinek előzetes írásbeli hozzájárulása nélkül.



## A LEÍRÁSBAN HASZNÁLT FOGALMAK MAGYARÁZATA

E fejezetből megismerheti a program funkcióinak és kezelésének megértéséhez szükséges legfontosabb fogalmakat. A fogalmaknak megadjuk a számítástechnikában általában használt szinonímáit is és példákon mutatjuk be jelentésüket.

E fejezetet nem szükséges teljes mértékben áttanulmányozniuk a programozásban jártas felhasználóknak.

### ADATTÁR

*Szinonimák:* file, adatállomány, állomány, "MEMO"

*Jelentés:* Azonos szerkezetű SORokból álló adathalmaz.

Az adattárat neve és SZERKEZETE jellemzi. A számítógép memóriájában egyszerre csak egy adattár lehet. Lehetőség van arra, hogy a memóriában levő adattárat kiírjuk magnó kazettára és onnan azt később újra visszatöltsük a memóriába.

Az adattárnak kiírásakor nevet kell adnunk. Beolvasáskor e nevet kell megadnunk ahhoz, hogy a program tudja, melyik adattárat kívánjuk a kazettáról beolvasni.

Egy kazettán egymás után tehát több adattár is lehet. Ezek lehetnek ugyanolyan szerkezetű sorokból álló, de különböző adattartalommal felírt adattárak ill. eltérő szerkezetű adattárak. Annak érdekében, hogy könnyen megtaláljuk a kazettán az éppen szükséges adattárat célszerű nyilvántartani a kazetta tartalmát. A következőket célszerű feljegyezni:

- adattár neve
- felírás időpontja
- magnó számláló állása

*Példák:* címjegyzék — név, cím, telefonszám nyilvántartására  
könyvek nyilvántartása — szerző, cím, egyéb adatok  
dolgozók adatai — név, cím, személyi szám, stb.



## SOR

*Szinonímák:* Rekord

*Jelentés:* Egy sor 1–16 egymáshoz tartozó adatból, mezőből áll, melyek egy tárgyat, személyt stb. írnak le, jellemeznek attól függően, hogy a MEMO adattárat mire használjuk.

A sor szerkezetét mi határozhatjuk meg. Egy sor mezőinek hossza összesen legfeljebb 250 karakter (betű, számjegy, írásjegy) lehet.

*Példák:* Ha címjegyzék készítésére használjuk a MEMO-t, akkor egy sora lehet a következő:

Kovács József 1234 Budapest Alma u. 5. 333-444

Ez a sor 3 mezőből áll:

- a) Kovács József
- b) 1234 Budapest Alma u. 5.
- c) 333-444

Ha könyvek nyilvántartására használjuk a MEMO-t, akkor egy sor lehet a következő:

Babits Mihály összegyűjtött versei  
Szépirodalmi Könyvkiadó 1968 27 Ft

Ez a sor 5 mezőből áll:

- a) Babits Mihály
- b) összegyűjtött versei
- c) Szépirodalmi Könyvkiadó
- d) 1968
- e) 27 Ft



## MEZŐ

*Jelentés:* Egy adat, a sor megnevezett része, melyet neve, típusa és hossza jellemez.

A mező neve legfeljebb 10 betűből ill. számjegyből álló karakter sorozat. A program használata közben nevének megadásával hivatkozhatunk a mezőkre, a listákon is gyakran szerepel. A hivatkozás-kor nem kell mindig begépnünk a teljes mezőnevet, elég az elejéről annyi karakter, ami már egyértelműen azonosítja a mezőt. Ha pl. a sor mezőinek neve név, cím, telefon, a „n” betű, mint a „név” szó első betűje egyértelműen megmondja a programnak, hogy nem a cím és nem a telefon nevű mezőre gondoltunk, hanem a név mezőre.

A mező típusa azt fejezi ki, hogy milyen adatot tárolunk a mezőben. A MEMO-ban kétféle mezőtípust használhatunk.

Numerikus (n) mezőben egész vagy tizedestört alakban számokat tárolhatunk. Ha a számokból álló mező tartalmát összegezni kívánjuk a MEMO-val, akkor feltétlenül numerikusnak definiáljuk a mezőt.

Az alfanumerikus (a) mezőben tetszőleges betűkből, számokból ill. egyéb jelekből álló karaktersorozatot tárolhatunk.

A mező hossza legfeljebb 52 karakter lehet. Adatbevitelkor legfeljebb ennyi karaktert (betű, szám, stb.) gépelhetünk be a mezőbe, ezért becsüljük meg előre a várható maximális mezőhosszt.

*Példák:* A sor meghatározásánál adott első példában a mezők jellemzői a következők:

mező neve	típusa	hossza
név	a	19
cím	a	30
telefon	a	10



A sor meghatározásánál adott második példában a következőképpen definiálhatjuk a mezőket:

<u>mező neve</u>	<u>típusa</u>	<u>hossza</u>
szerző	a	15
cím	a	20
kiadó	a	15
év	a	4
ár	n	7



## JELZŐ

*Jelentés:* Minden sornak két jelzője lehet:

a „t” törlés jelző és

a „\*” kiválasztás jelző.

Alaphelyzetben, a sor adatainak begépelése után a sor egyik jelzővel sincs megjelölve.

A „t” jelzővel azokat a sorokat jelöljük meg, melyeket ki akarunk hagyni, ki akarunk törölni az adattárból.

A „\*” jelzővel az adattár egy bizonyos tulajdonságú sorait jelölhetjük meg.

*Példák:* Ha ki akarjuk hagyni adattárunkból a korábban példaként megadott sort, a sor kiválasztása után a „t” paranccsal beállíthatjuk a sor törlés jelzőjét.

A beállítás után a sor a következőképpen néz ki:

t Kovács József 1234 Budapest Alma u. 5. 333-444

A törlés jelző beállításával még nem vészett el a sor, a törlést ténylegesen a „kihagy” parancs hajtja majd végre.

A „j+ név<L” parancs kijelöli azokat a sorokat, amelyben a név mező első betűje az abc-ben az L előtt szerepel. A kijelölés után a korábban említett példa sor így néz ki:

\* Kovács József 1234 Budapest Alma u. 5. 333-444

Természetesen az is előfordulhat, hogy a sor mindkét jelzője be van állítva, pl.:

\* t Kovács József 1234 Budapest Alma u. 5. 333-444



## SORSZÁM

*Szinonímák:* Rekord sorszám, azonosító szám

*Jelentés:* Az adattár minden sora sorszámot kap, mely azt jelzi, hogy a sor hányadik az adattár elejétől. A sorszám a sor fontos azonosítója. Vigyázni kell, mert egy adatsor sorszáma megváltozhat, ha a sor elé beszúrunk egy vagy több új sort.

A sorszámok között az egyiknek kitüntetett jelentősége van, ez az ún. aktuális sor száma. Felhívjuk a figyelmet arra, hogy bizonyos parancsok eredménye eltérő, ha a parancs kiadásakor másik sor az aktuális.

Pl. a „lista” parancs az aktuális sortól listázza az adattár sorait, így ha az aktuális sor száma nem 1, akkor nem listázza ki az összes sort. Természetesen parancsok segítségével megváltoztathatjuk az aktuális sor számát és bármikor megkérdezhetjük, hogy éppen melyik sor az aktuális.

*Példák:* Adattárunk tartalmazzon 5 sort, egy sor álljon két mezőből (megnevezés, ár):

sorszám	megnevezés	ár
1	krumpli	8
2	könyv	65
3	kazetta	140
4	kifli	5
5	bérlet	45

Az adatok beírása után, ha a kazetta adatait akarjuk módosítani, gépeljük be a parancs sorban a 3-as számot. Ezután a 3 lesz az aktuális sor száma. Erről a „?” parancssal győződhetünk meg.



## A MEMO parancsai

A MEMO programot betöltése után parancsokkal irányíthatjuk. Egy-egy parancs egy-egy funkciót hajt végre. Pl. a „BELR” parancs segítségével írhatunk be új adatokat az adattárba, a „LISTÁZ” parancssal listázhatjuk ki a képernyőre az adattárat.

A parancs kiadásához be kell gépelnünk a parancs nevét és esetleg néhány kiegészítő adatot a képernyő 23. sorának elején a szaggatott vonallal jelölt részen. A „LISTÁZ” parancs esetében, ha pl. nem akarjuk a sor minden mezőjét kilistázni, kiegészítő adatként kell megadni a listázandó mezők nevét. A parancs neve és az őt esetleg követő mezőnév közé legalább egy szóközt (space) kell gépelni.

A parancs gépelésének befejezését a „RETURN” billentyű lenyomásával jelezzük.

Nem szükséges begépelni a parancs ill. mező neveket teljes hosszúkbán.

A parancs nevéből elég megadni azt az első egy-két betűt, melyet a leírásban kisbetűvel közlünk. Ezeket a betűket legegyszerűbben kisbetűkkel gépelhetjük be, hiszen a számítógép billentyűzete alaphelyzetben, a SHIFT billentyű használata nélkül ilyen betűket generál.

Pl. a LISTÁZ parancs nevéből elég begépelni az első két betűt (li), a reNDEZ parancs kiadásához elég megadnunk az (re) betűket.

*Vigyázat:* A parancsokat kisbetűkkel kell megadni!

A mezőnevekben a kis- és nagybetűk nem cserélhetők fel! Pl. Cím helyett nem lehet cím.

A mezők nevéből elég a név elejéről annyi karaktert megadni, amennyi egyértelműen azonosítja a mezőt.

Ha pl. a sor mezőinek neve: név, azonosító, nem, kor, képesítés, a név mezőt egyértelműen kijelöli az első két betűje (né), mivel a másik n betűvel kezdődő mezőnévben a második betű már eltérő (é helyett e). Ugyanebben a példában az azonosító nevű mező esetén elég megadni az első betűt, mivel nincsen másik (a) betűvel kezdődő mezőnév a sorban.

A parancsok nevét könnyen meg lehet tanulni, hiszen az általa megvalósított funkciót fejezi ki. Amennyiben elfelejtenénk valamelyik parancs nevét a „seGIT” parancs hatására kiíródik az összes parancs neve a képernyőre emlékeztetőül. A parancsok leírását e fejezet további része tartalmazza a „seGIT” parancs által kiírt sorrendben.



## PARANCS MENÜ

*Parancs:* seGÍT

*Funkció:* A MEMO parancsai jelennek meg a képernyőn a következő formában:

### ADATBEVITEL

beIR  
móDOSÍT

### KIJELÖLÉS

jf  
j+ <mezőnév> <jel> <érték>  
j\* <mezőnév> <jel> <érték>  
foRDÍTVA  
\* -

### TÖRLÉS

t  
\*t  
kiHAGY  
t-  
-----  
ösSZEGEZ <mezőnév>  
\*ö <mezőnév>

### LISTÁZ

liSTÁZ <mezőnevek>  
\*l  
\*l <mezőnevek>  
nyOMTAT  
nyOMTAT <mezőnevek>  
\*n  
\*n <mezőnevek>  
?  
<szám>  
+<szám>  
-<szám>  
<RETURN> vagy + (+1)  
^ vagy - (-1)

reNDEZ <mezőnév>

-----  
szERKEZET  
töRLI (a képernyőt)  
feLVESZ (a magnóra)  
újRA (kezdés)  
véGE

*Megjegyzés:* A parancs mezőben automatikusan megjelenik a segít parancs név, ha nem létező parancs nevet írunk be. Ebben az esetben elég megnyomni a <RETURN> billentyűt, ha segítségképpen ki akarjuk iratni a fenti képernyőt. Ellenkező esetben be kell gépelni a kívánt parancs nevet.



## ADATBEVITEL-MÓDOSÍTÁS

*Parancs:* beLR

*Funkció:* Új adatok beírása az adattárba. A parancs segítségével egy vagy több sort illeszthetünk be az aktuális sor után.

Az új sor egyes mezőinek értékét, tartalmát párbeszédés formában kell megadni. Ez azt jelenti, hogy a program kiírja a mező nevét és egy kettőspontot. A mező értékét a kettőspont után kell begépelni, majd a „RETURN” billentyű lenyomásával jelezzük, hogy nem kívánunk több karaktert gépelni a mezőbe.

Ha csak a „RETURN”-t ütjük le, akkor numerikus mező esetén annak értéke 0 lesz, alfanumerikus mező esetén a mező üres marad. A mező gépelése közben természetesen a program a mező típusának és hosszának megfelelő tartalmat enged csak begépelni. Numerikus mezőbe csak egy egész vagy tizedestört alakú szám gépelhető be (tizedes pontot kell használni!).

Csak annyi karakter gépelhető a mezőbe, amilyen hosszúra definiáltuk a mezőt előzőleg.

Ha begéveltük egy sor összes mezőjét, a program újabb sor adataira vár. Ha nem kívánunk több sort begépelni, nyomjuk le a ^ billentyűt a sor első mezőjének megadása helyett. Ezután a program újabb parancsra vár.

Ha a sorok beírásakor az adattár megtelik, de még lehetséges lenne az adattár növelése, akkor a „Növelje a maximális sorszámot” üzenet megjelenése után a következőképpen lehet eljárni:

- A feLVESZ paranccsal kimentjük az adattárat kazettára. A kimentés elején lefolytatott párbeszédben a program „Kívánja-e növelni a megadható sorok számát?” kérdésére az „i” (igen) választ adjuk, majd a „Hány sor legyen?” kérdésre adjuk meg az új maximális sorszámot.
- A mentés befejezése után a programot az újRA parancs segítségével újra elindíthatjuk és az előbb kimentett adattárat



újra beolvashatjuk a memóriába. Ekkor már annyi sor fér az adattárba, amennyire definiáltuk a kimentéskor. Persze a sorok száma nem növelhető tetszőleges mértékig, mert ezt a sor hossza és a memória méret behatárolja (lásd MEMO korlátai c. fejezetet).

Új adattár definiálása után (ha a program betöltése, vagy az újRA nevű paranccsal való újraindítás után a program kérdésére azt válaszoljuk, hogy új adattárral kívánunk foglalkozni és definiáljuk az új adattár szerkezetét) először a beR parancsot kell használni, hiszen ekkor az adattár még üres, így más parancsok nem nagyon van értelme.

*Példa:* Írjunk be egy új sort a „könyvek” nevű adattárba, amikor az aktuális sor száma 1:

be<RETURN>

2. sor

Szerző : Radnóti Miklós<RETURN>

Cím : Összes versei<RETURN>

Ár : 56.5<RETURN>

3. sor

Szerző : ^



## *Parancs: módOSÍTÁS*

*Funkció:* Az aktuális sor tartalmának módosítása.

A beíR paranccsal korábban rögzített adatokat módosíthatjuk ezzel a paranccsal. E parancs használata hasonlít a beíR parancshoz. A parancs kiadás után megjelenik a képernyőn az aktuális (kiválasztott) sor, a mező nevek után az eredeti, módosítani kívánt adatok látszanak.

Egymás után végigmegyünk az összes mezőn. Ha módosítani akarjuk a mezőt, akkor beírjuk az új értéket majd megnyomjuk a RETURN billentyűt. Vigyázni kell, mert a régi karaktereket felül kell írni, ki kell cserélni. A mező új értéke a RETURN lenyomásakor látott tartalom lesz. Ha az egyik mező értéke megfelelő, nem akarjuk változtatni, akkor csak a RETURN billentyűt nyomjuk le.

Mivel e parancs az aktuális sor módosítására alkalmas, ezért használata előtt ki kell választanunk a módosítani kívánt sort. Ezt általában a ?, <szám>, +<szám> ill. -<szám> parancsok segítségével tehetjük meg.

*Példa:* Módosítsuk a beíR paranccsnál megadott példában beírt sor szerzőjét Babits Mihályra. Ha a módosítani kívánt sor az aktuális sor, akkor a következő karaktersorozat leütésével érhetjük ezt el:

mó<RETURN>Babits Mihály <RETURN><RETURN><RETURN>



## KIJELÖLÉS

*Parancs:* jf

*Funkció:* Az aktuális sor \* jelzőjét mindenféle feltétel vizsgálata nélkül az ellentétes értékre állítja, vagyis a sor meg lesz jelölve \*-gal, ha eddig nem volt, ill. a \* kiválasztás jelző törlődik, ha eddig szerepelt.

*Példa:* Legyen az aktuális sor a belR parancsnál példaként megadott sor. Adjuk ki a jf parancsot a „jf<RETURN>” billentyűsorozat leütésével. Írassuk ki az aktuális sort a „?” parancssal. Láthatjuk, hogy a sor meg lett jelölve a \*-gal. Most újból adjuk ki a „jf” parancsot és utána a „?” parancsot. Láthatjuk, hogy a \* jelző eltűnt.

*Parancs:* j+ <mezőnév> <összehasonlítójel> <érték>

*Funkció:* A parancs hatására \* jelzőt kapnak azok a sorok, melyekre a <mezőnév> <összehasonlítójel> <érték> feltétel teljesül. Vagyis kijelölhetünk olyan sorokat, melynek egy bizonyos mezőjének értéke megadott tulajdonságú. A parancs nevet legalább egy szóköznek valamint egy feltételnek kell követnie. A <mezőnév> a sor egyik mezőjének neve. Itt is elég megadni a mezőnév egyértelmű kezdetét.

Az <összehasonlítójel> (reláció) hatféle lehet:

- < kisebb
- = egyenlő
- > nagyobb
- <= kisebb vagy egyenlő
- <> nem egyenlő
- >= nagyobb vagy egyenlő

Az <érték> a megnevezett mező típusának megfelelően szám vagy tetszőleges karaktersorozat lehet. Alfanumerikus típusú mező esetén az összehasonlítás csak a megadott karaktersorozat (betűsorozat) hosszában történik.



*Példa:* A j+ Szerző<K parancs megjelöli azokat a sorokat, amelyekben a „Szerző” nevű alfanumerikus típusú mező első betűje a magyar abc-ben a K betű előtt szerepel.

A j+ Ár>100 parancs megjelöli azokat a sorokat, amelyekben az „Ár” nevű numerikus mező értéke 100-nál nagyobb (100 forintnál drágább könyvek pl.).

*Parancs:* j \* <mezőnév> <összehasonlítójel> <érték>

*Funkció:* E parancs hasonló funkciójú és formájú az előző j+ paranccsal. Az eltérés az, hogy a j \* parancs csak azok közül a sorok közül jelöli meg a feltételnek megfelelőket, melyek korábban a parancs kiadása előtt már meg voltak jelölve. Tehát ha ki vannak jelölve az „A” feltételnek megfelelő sorok és kiadjuk a j \* <B> parancsot, akkor azok a sorok maradnak megjelölve, melyekre az <A és B> feltétel teljesül.

A parancs nevét követően ugyanolyan formájú feltételt kell megadni, mint a j+ parancs esetén.

Ha a korábban \* -gal kijelölt sor teljesíti a megadott feltételt, akkor továbbra is megmarad a \* jelzője.

Ha a sor nem teljesíti a megadott feltételt, akkor nem lesz \* jelzője akkor sem, ha korábban ki volt jelölve és akkor sem, ha korábban sem volt kijelölve.

*Példa:* Jelöljük ki a j+ parancsnál példaként megadott mindkét feltételnek megfelelő sorokat:

j+ Szerző<K<RETURN>

j \* Ár>100<RETURN>

A két parancs együttes hatása az, hogy azok a sorok lesznek megjelölve, amelyek szerzője az abc szerint K előtt szerepel és amely könyv ára több mint 100 forint.



*Parancs:* foRDITVA

*Funkció:* Az adattár összes sorának \* jelzőjét ellentétes értékre állítja, vagyis \* jelzőt kap az a sor, amelyiknek eddig nem volt és a korábbi \* jelzők törlődnek.

*Példa:* Jelöljük ki a j+ Ár>100 paranccsal a 100 forintnál drágább könyveket majd adjuk ki a fo parancsot. Ekkor a 100 forintnál nem drágább könyvek lesznek kijelölve, vagyis azok, amelyek vagy olcsóbbak 100 forintnál vagy éppen 100 forintba kerülnek.

*Parancs:* \* –

*Funkció:* Az összes sor \* jelzőjét alaphelyzetbe állítja, kitörli. Ez a parancs akkor hasznos, amikor korábban már kijelöltünk bizonyos feltételeknek megfelelő sorokat, de most egy másik, az előzőtől független feltétel szerint akarunk adatokat keresni.

*Példa:* Jelöljük ki sorokat a j+ Ár>100 paranccsal. Győződjünk meg arról, hogy a megfelelő sorok valóban meg lettek jelölve \*-gal (li parancs). Adjuk ki a \* – parancsot, majd nézzük meg, hogy a \* jelzők ki lettek törölve (li parancs).



## TÖRLÉS

*Parancs:* t

*Funkció:* A kiválasztott, aktuális sor t törlésjelzőjét beírja. Ezzel a paranccsal jelölhetünk ki törlésre egy-egy rekordot, melyekre a későbbiekben már nem lesz szükségünk.

*Példa:* Legyen a címjegyzék aktuális sora a következő:

Jelzők:  
név : Kovács József  
cím : 1234 Bp. Alma u. 27  
telefon : 345-678

Adjuk ki a t parancsot és irassuk ki újra a sort a ? paranccsal:

Jelzők: t  
név : Kovács József  
cím : 1234 Bp. Alma u. 27  
telefon : 345-678

Láthatjuk, hogy a sor tartalma nem változott, de meg lett jelölve a t jelzővel.

*Parancs:* \*t

*Funkció:* A \* jelzővel kijelölt sorok t törlésjelzőjének beállítása. Ezt a parancsot akkor érdemes használni, amikor több, adott tulajdonságú sort akarunk kijelölni törlésre.

Ebben az esetben először a kijelölésre szolgáló parancsokkal (jf, jt, j\*, fo) beállítjuk a megfelelő sorok \* jelzőjét, majd kiadjuk a \*t parancsot.

*Példa:* Jelöljük ki törlésre az 50 forintnál olcsóbb könyveket:

```
j+ Ár<50<RETURN>  
*t<RETURN>
```

Listázzuk ki az adattárat, hogy meggyőződjünk a helyes eredményről:

```
li<RETURN>
```



*Parancs:* kiHAGY

*Funkció:* A t törlésjelzővel törlésre kijelölt sorok tényleges kitörlése az adattárból.

Ezt a parancsot akkor kell kiadni, ha előzőleg a t ill. \*t parancsokkal beállítottuk bizonyos sorok törlésjelzőjét, megbizonyosodtunk róla, hogy a megfelelő sorokat jelöltük ki és végre akarjuk hajtani a törlést.

A kiHAGY parancs végrehajtása után a kitörölt sorok elvesznek a memóriából, többé nem lesznek felhasználhatók.

*Példa:* Jelöljük ki törlésre a 80 forintnál olcsóbb könyveket és hajtsuk végre a törlést:

```
j+ Ár<80<RETURN>
*t<RETURN>
ki<RETURN>
```

*Parancs:* t-

*Funkció:* Az adattár összes sora törlésjelzőjének alaphelyzetbe állítása.

Ezt a parancsot akkor célszerű használni, ha a t ill. \*t parancsokkal kijelöltünk bizonyos sorokat törlésre, de mégsem akarjuk kitörölni e sorokat.

*Példa:* Jelöljük ki azokat a sorokat, melyek szerzőjének neve G-vel vagy G utáni betűvel kezdődik. Állítsuk vissza alaphelyzetbe a törlésjelzőket és listázzuk ki a sorokat, hogy meggyőződjünk a parancs helyes végrehajtásáról:

```
j* Szerző=G<RETURN>
*t <RETURN>
li <RETURN>
t- <RETURN>
li <RETURN>
```



## ÖSSZEGZÉS

*Parancs:* `ösSZEg <mezőnév>`

*Funkció:* Az adattár sorok megadott mezőjének összegzése.

A parancsban szereplő mezőnév csak numerikus típusú mező neve lehet. A parancs kiszámítja a kijelölt mezők összegét és kiírja az összeget a képernyőre.

*Példa:* Összegezzük az ár mezők tartalmát:

```
ös Ár<RETURN>
```

*Parancs:* `*össZEg <mezőnév>`

*Funkció:* E parancs funkciója hasonló az `ösSZEg` parancséhoz, az eltérés az, hogy csak a \* jelzővel kijelölt sorok mezőit összegzi, nem az összes sorét.

*Példa:* Összegezzük az M betűs szerzők könyveinek árát:

```
j+ Szerző=M<RETURN>
```

```
*ö Ár<RETURN>
```



## LISTÁZÁS

*Parancs:* liSTA <mezőnév> <mezőnév> . . .

*Funkció:* Az adattár sorainak a képernyőn való megjelenítése az aktuális sortól az utolsó sorig.

A parancsnév után szóközökkel vagy vesszőkkel elválasztva megadhatunk mezőneveket. Ekkor a program csak a megadott mezőket írja ki a képernyőre a sorokból. Egy-egy mező neve megadható többször is, azonban összesen legfeljebb annyi mezőt adhatunk meg, ahány mezőből áll a sor.

Ha a megjelenítendő sor képe hosszabb lenne, mint 64 betűhely (az elválasztás céljából beszúrt szóközökkel és a jelzőkkel együtt), akkor a MEMO a sort mezőnként új képernyő sorba tördelve jeleníti meg. Ilyenkor minden sor adatai előtt külön sorban szerepelnek a sor jelzői (\* ill. t) és minden mező előtt megjelenik a neve is.

Ha a sor tartalma elfér egy képernyő sorban, akkor a mezőnevek a képernyő 24. sorában jelennek meg emlékeztetőül. Ha egy mező hossza rövidebb a saját nevénel, akkor nevének csak a kezdete jelenik meg.

*Példa:* Ha a megjelenítendő sor hossza kisebb, mint 64 a címjegyzék egy sora a li<RETURN> parancs hatására a következő formában íródik ki a képernyőre:

\* Kovács József 1234 Bp. Alma u. 27. 345-678

Ha a sor nem fér ki a képernyő egyetlen sorába, akkor a li<RETURN> parancs a következő formában írja ki:

Jelzők: \*

név : Kovács József  
cím : 1234 Bp. Alma u. 27.  
telefon : 345-678

Listázzuk ki a könyvek adattárban nyilvántartott szerzőket:

li Szerző <RETURN>



*Parancs:* \*IISTA <mezőnév> <mezőnév> ...

*Funkció:* Ennek a parancsnak a funkciója hasonló a már ismertetett IISTA parancshoz. A \*IISTA parancs azonban csak a \* jelzővel kijelölt sorok tartalmát jeleníti meg a képernyőn.

Ezt a parancsot akkor célszerű használni, ha a kijelölő parancsokkal (jf, jt, j\*, fo) kijelöltünk bizonyos tulajdonságú sorokat és kíváncsiak vagyunk azok tartalmára.

*Példa:* Listázzuk ki a K betűs szerzők könyveit adattárunkból:

```
j+ Szerző=K <RETURN>
*I <RETURN>
```

## NYOMTATÁS

*Parancs:* nyOMTAT <mezőnév> <mezőnév> ...  
\* nYOMTAT <mezőnév> <mezőnév> ...

*Funkció:* A IISTA parancshoz hasonló nyOMTAT parancsokkal sornyomatón is kiíráthatjuk az adattár sorait.

A nyOMTAT ill. \*nyOMTAT parancs az aktuális sortól kezdve az adattár végéig kinyomtatja az adattár összes ill. csak a \*-gal kijelölt sorát.

A nyomtatási kép is hasonló ahhoz, amit a IISTA parancsoknál a képernyőn látunk. A papíron azonban 80 betűhely áll rendelkezésre, így hosszabb sorok jeleníthetők meg tördelés nélkül.

E parancsok természetesen csak akkor használhatók, ha a TV Computerhez a beépített csatlakozón keresztül nyomtató van kapcsolva.

*Példa:* Nyomtassuk ki a teljes adattárból a cím és ár mezőket:

```
ny Cím Ár<RETURN>
```



## SOR KIVÁLASZTÁS

*Parancs:* ?

*Funkció:* Az aktuális sor megjelenítése a képernyőn.

A leírásban használt fogalmak magyarázatánál már szoltunk az aktuális sor jelentéséről. A parancsok egy része más eredményt ad eltérő aktuális sorok esetén, ezért fontos az ilyen parancsok kiadása előtt meggyőződni arról, hogy a helyes sor az aktuális. Erre szolgál a ? parancs.

*Példa:* Irassuk ki az aktuális sort: ? <RETURN>

*Parancs:* <szám>

*Funkció:* A megadott számú sor kiválasztása.

A parancs kiadása után (egy szám beírása után lenyomjuk a RETURN billentyűt) a begépelte számú sor lesz az aktuális sor és a sorszám megjelenítődik a 24. sorban.

*Példa:* Legyen a 10. sor az aktuális: 10<RETURN>

Legyen a 16. sor az aktuális: 16<RETURN>

*Parancs:* +<szám>

-<szám>

*Funkció:* E két parancs szintén az aktuális sor megváltoztatására szolgál.

A +<szám> parancs hatására előre,

a -<szám> parancs hatására visszafelé lépünk annyi sort, amennyi a megadott szám.

A következő sort tehát a +1, az előző sort a -1 megadásával választhatjuk ki.

A +1<RETURN> helyett elegendő csak a <RETURN> billentyűt megnyomni vagy a +<RETURN> begépelése.

A -1 parancs (előző sor kiválasztása) a ^ vagy a -<RETURN> billentyűk leütésével adható ki egyszerűbben.

*Példa:* A 10. sor kiválasztása után lépünk előre 2 sort a 12. sorra:

10<RETURN>

+2<RETURN>



Lépjünk előre 1 sort:

<RETURN>

Lépjünk visszafelé 1 sort:

^

## RENDEZÉS

*Parancs:* reNDEZ <mezőnév>

*Funkció:* Az adattár sorainak átrendezése a megadott mező értékei szerint.

Ha a rendezés szempontjául megadott mező numerikus típusú, akkor a rendezés után a sorok sorrendje olyan, hogy e mező értéke növekszik.

Ha a rendezés kulcsául megadott mező alfanumerikus típusú, akkor a rendezés során a sorok sorrendje olyan, hogy e mező a magyar abc szerinti sorrendben szerepel. A rendezés során kis és nagybetű között a MEMO nem tesz különbséget.

A rendezés után az adattár első sora lesz az aktuális (kiválasztott) sor.

A rendezéssel természetesen megváltozik a sorok sorszáma.

A reNDEZ parancs által végrehajtott funkció a legösszetettebb a MEMO parancsai között. A parancs végrehajtásának ideje függ a rendelkezésre álló szabad memória terület nagyságától. Ezért ne válasszuk feleslegesen túl nagyra az adattár sorainak számát.

*Példa:* A címjegyzékbe felvettünk különféle címeket a beÍR paranccsal. Rendezzük a címeket a név szerint abc sorrendbe: re név <RETURN>

Listázzuk ki az adatokat a rendezett sorrendben:

li<RETURN>

Rendezzük a könyvek nevű adattárat a könyvek ára szerint:

re Ár<RETURN>

A rendezés után az első sor tartalmazza a legolcsóbb, az utolsó sor a legdrágább nyilvántartott könyvet.



## EGYÉB PARANCSONK

*Parancs:* szERKEZET

*Funkció:* A sorok szerkezetének és az adattár pillanatnyi állapotának megjelenítése a képernyőn.

A sor szerkezet táblázatban íródik ki a képernyőre, a táblázat oszlopai: a mező neve, típusa, hossza.

Az adattár állapotára vonatkozó információk: a sorok száma, a megjelölt sorok száma, a sorok számának lehetséges maximuma.

*Példa:* Irassuk ki a memóriában tárolt adattár szerkezetét:

```
sz<RETURN>
```

*Parancs:* törLI

*Funkció:* A képernyő törlése.

Ha a parancsok végrehajtása közben a képernyő betelik, nincs már üres sor a képernyő alján, a legfelső adatsor eltűnik, az alatta levő sorok egy-egy sorral feljebb kerülnek és így egy üres sor képződik a kiírandó adatok számára. Ezzel a módszerrel a MEMO program mindig csinál helyet a megjelenítendő adatoknak. Sok esetben azonban jobban áttekinthetővé válnak az adatok, ha a régebben kiírt adatokat töröljük a parancs előtt a képernyőről. Ezt a célt szolgálja ez a parancs.

*Példa:* Listázzuk ki az adattárat, a megjelenített adatok tanulmányozása után töröljük a képernyőt, majd irassuk ki az ár nevű mezők összegét a legfelső adatsorban:

```
li <RETURN>  
tö <RETURN>  
ös Ár<RETURN>
```



**Parancs:** feLVESZ

**Funkció:** A memóriában tárolt adattár kiírása, elmentése kazettára.

A parancs kiadása után a program megkérdezi az állomány nevét. A parancs lehetővé teszi, hogy növeljük az adattár méretét (sorok száma) a lehetséges maximumig. A kimentett adattár legközelebbi beolvasásakor már a módosított adattár méret lesz érvényes.

A kazettás magnó kezelését párbeszéddel irányítja a program, mindig előre közli a teendőket.

A parancs kiadása előtt azonban nekünk kell gondoskodnunk arról, hogy a megfelelő kazetta legyen a magnóban és a kazettában a szalagnak azon a helyén álljunk, ahol nem írunk felül értékes programot vagy adatot.

Érdemes az adattáron végzett műveletek előtt már kikeresni a megfelelő kazetta megfelelő pontját, ahova a későbbiekben nyugodtan felvehetjük az adattárat.

A kazettán tárolt adatokról célszerű nyilvántartást vezetni (lásd az ADATTÁR fogalom leírását).

**Példa:** Az újonnan létrehozott adattárba a beÍR paranccsal sok értékes adatot gépeltünk be, listázással meggyőződünk arról, hogy az adatok gépelése közben nem hibáztunk (esetleges hibákat a módOSIT paranccsal javíthatjuk).

Az adatok feldolgozása előtt (kijelölések, rendezés, összegzés) kimentjük kazettára az adattárat: fe<RETURN>

**Parancs:** újRA

**Funkció:** A MEMO program újraindítása anélkül, hogy a programot újra be kellene tölteni kazettáról. A parancs végrehajtásakor a memóriában tárolt eddigi adatok törlődnek!

Ezt a parancsot akkor célszerű használni, ha nincs már szükségünk a memóriában levő adattárra (ha értékes adatokat tartalmaz, akkor felvettük kazettára) és egy másik adattárral akarunk a továbbiakban foglalkozni. Ezt az új adattárat betölthetjük kazettáról – ha már régebben létrehoztuk – vagy új adattárat definiálhatunk.

Az előzőekben vázolt funkciókat a következő lépésekben hajthatjuk végre a program által irányított párbeszéd segítségével:



A program megkérdi, hogy az új adattárat a kazettáról olvassa-e be, vagy pedig új adattárat kívánunk felépíteni: Régi adatokkal kíván foglalkozni? (i/n)

Általában régi adatok ismételt felhasználására kerül sor. Ekkor a program megkérdi az adattárat tartalmazó kazettás állomány nevét és a megfelelő üzenetek és várakozások segítségével az adattárat megkeresi a kazettán, majd beolvassa. A beolvasás után megjeleníti az adattár jellemzőit és parancsra vár.

Ha a megfelelő kérdésre adott válaszból az derül ki, hogy új adattárat kívánunk létrehozni, akkor a MEMO rátér a sorok szerkezetét meghatározó párbeszédre.

Ennek során meg kell adni a sor mezőinek nevét, típusát és hosszát. Az utolsó mező megadása után a következő mező neve helyett a ^ billentyűt kell leütnünk, ez jelzi a programnak a szerkezet definiálásának végét.

Ezután megjelennek az adattár eddig meghatározott jellemzői és a MEMO közli, hogy a megadott tulajdonságú sorokból legfeljebb mennyivel tud foglalkozni (ezt a memória mérete korlátozza). Ekkor a felhasználónak lehetősége van arra, hogy a maximális szám helyett egy kisebb értékben határozza meg az adattár sorainak számát. Ez ajánlatos is, mert a feleslegesen fenntartott helyek lassítják a MEMO működését. Az adattár kazettára való felvétele során egyébként ezt a pillanatnyilag lecsökkentett méretet szükség esetén meg tudjuk növelni (lásd felVESZ parancs).

Ha az adattár-meghatározó párbeszéd befejeződik, akkor a MEMO készen áll az első sor befogadására: a program olyan állapotban van, mint más esetekben a beIR parancs hatására. Ha több sort már nem akarunk bevinni, akkor a következő sor első mezője helyett csak a ^ billentyűt nyomjuk meg. Ennek hatására a MEMO parancsra vár, amit a képernyő 23. sorának elején villogó kurzor jelez.

Megjegyzés: Új adattár létrehozása után közvetlenül, amíg legalább egy sor adatot nem rögzítettünk, csak a következő parancsok használhatók: beIR, szERK, véGE, újRA.



*Parancs: véGE*

*Funkció:* A MEMO program futtatásának végét jelezhetjük ennek a parancs-nak a kiadásával.

Ha az utolsó feLVESZ parancs óta módosítottuk a memóriában tárolt adattárat, akkor egy párbeszéd következik a program és a felhasználó között, amelynek során rendelkezhetünk arról, hogy ki akarjuk-e írni kazettára az adattárat (csak elfelejtettük korábban kimenteni a feLVESZ paranccsal).



## HOGYAN HASZNÁLJUK A MEMO PROGRAMOT?

### Általános tudnivalók

A MEMO párbeszédés program. Ez azt jelenti, hogy használatát bizonyos mértékben maga irányítja. A párbeszédet magyar nyelven folytatja. Felhívjuk a figyelmet az ékezetes betűk helyes használatára! Pl. mivel az u és ú betűk különböznek egymástól, ezért az újRA parancsot nem adhatjuk ki az újRA<RETURN> formában.

A program a képernyő különböző részein jeleníti meg üzeneteit és a felhasználó is a képernyő különböző részein válaszol.

A program akkor vár válaszra, ha a képernyőn a villogó téglalapocskát (az ún. kurzort) látjuk. Ilyenkor vagy egy számot vagy egy szöveget (betűk, számok, jelek sorozata) kell begépelni. A legtöbbször ezen a helyen már van valami. Ha a felhasználó éppen ezt szeretne volna begépelni, akkor elegendő a <RETURN> billentyűt lenyomni. A begépelhető szöveg helyét a MEMO aláhúzás jelekkel tölti ki, jelöli (ha még nincsen szöveg). Ha az utolsó beírható hely után még további betűket írunk, akkor az felülírja az utolsó betűt.

Ha a gépelést elrontottuk (vagy csak módosítani akarjuk a MEMO által kiírt szöveget), akkor a kurzor mozgató billentyűkkel ill. a botkormánnyal jobbra-balra mehetünk a megfelelő helyig és a többi szövegszerkesztő billentyű segítségével is a TV Computeren megszokott módon javíthatjuk a szöveget. A <RETURN> billentyű lenyomása azt jelzi, hogy a szövegen további módosítást nem kívánunk végrehajtani.

A következő szövegszerkesztő billentyűket lehet használni:

Billentyű	Funkció
DEL (delete)	A kurzor előtti jelet törli.
INS (insert)	Új betűhelyet szúr be.
ESC (escape)	A mező elejére viszi a kurzort.
SHIFT-DEL	A kurzor helyén levő jelet törli.
CTRL-k	Törli a mezőnek a kurzortól jobbra levő részét.
CTRL-y	Az egész mezőt törli.



A MEMO néhány párbeszéde során a ^ billentyű leütésének különleges hatása van: vissza lehet lépni az előző párbeszéd elemre. Ha a párbeszéd első válasza helyett ütjük le a ^ billentyűt, az azt jelenti, hogy nem kívánjuk lefolytatni a párbeszédet. Ha a ^ billentyűt nem ilyen helyen ütjük le, akkor hatása a <RETURN> billentyűével azonos.

A MEMO a begépett információt a <RETURN> hatására kezdi feldolgozni. Ez alól csak azok az esetek a kivételek, amikor csak egyetlen billentyű leütésére vár a program.

A MEMO a parancsokat a képernyő 23. sorának első 32 betűhelyén várja, a hibajelzéseket pedig a képernyő 24. sorának második 32 betűhelyén írja ki.

## A PROGRAM INDÍTÁSA

A programot a kazettáról a

```
load "memo" <RETURN>
```

paranccsal tölthetjük be. (Ha a kazetta a program elején áll, akkor nem szükséges a program nevet megadni, elegendő a load<RETURN> parancs.)

Sikeres betöltés után a program automatikusan elindul, kiírja a program nevét tartalmazó címlapot és várja, hogy a felhasználó leüsse a szóköz (space) billentyűt.

Ekkor a program ugyanabban az állapotban van, mintha az újRA parancsot adtuk volna ki. Lehetőségünk van arra, hogy új adattárat építsünk fel vagy régi adattárat olvassunk be kazettáról.

A további teendőket és lehetőségeket az újRA parancs leírása tartalmazza.



## PÉLDA A MEMO HASZNÁLATÁRA

Az alábbi példa a gép előtt ülve a MEMO programot tartalmazó kazettán levő példa adatállomány segítségével végigpróbálható.

A baloldali oszlopban az ön által beírandó szövegeket adjuk meg, a jobboldalon pedig rövid magyarázatot találunk.

load "memo" <RETURN>	A MEMO betöltése.
i <RETURN>	Igen, kazettáról akarunk olvasni.
konyvek<RETURN>	Az állomány neve a kazettán, Magnó indítás ill. leállítás a program utasítása szerint, szóköz leütése. Megjelenik az adattár szerkezete.
1<RETURN>	Menjünk az első sorhoz.
lista<RETURN>	* Listát készítünk a képernyőre.
<RETURN>	Folytassuk a listázást.
*-<RETURN>	A * jelzők alaphelyzetbe állítása.
j+ Cím<K<RETURN>	Jelöljük ki a K előtti című könyveket.
* lista<RETURN>	* Listázzuk ki a megjelölt sorokat.
j* Szerző>G<RETURN>	A megjelöltek közül válogassuk ki azokat, amelyek szerzője G-nél hátrább van a névsor- ban.
* lista<RETURN>	* Listázzuk ki a megjelölteket.
*összeg Ár<RETURN>	Mennyi a megjelölt könyvek ára összesen?
összeg Ár<RETURN>	Mennyi az összes könyv ára együtt?



* t<RETURN>	A megjelölteket jelöljük ki törlésre.
* lista<RETURN>	* Nézzük meg a jelzőket.
kihagy<RETURN>	Hagyjuk ki az adattárból a t-vel megjelölteket.
lista<RETURN>	* Megnézzük, mi maradt.
vége<RETURN>	Elköszönünk.
i<RETURN>	Igen, fel akarjuk venni kazettára a módosított adattárat.
kt<RETURN>	Ez az új állomány neve.
vége<RETURN>	Valóban befejezzük.

*Megjegyzés:* A magyarázat előtt \*-gal jelölt sorokban az adatállomány teljes vagy részleges listázását kérjük. A parancs sorozat helyes végrehajtása esetén a következő listáknak kell megjelenni a képernyőn. (A leírás elkészítésekor a nyOMTAT parancs segítségével készítettük el ezeket a listákat.)



* Joseph Heller	A 22-es csapdája	92
* J. R. Tolkien	A gyűrűk ura	149
* Dumas	A három testőr	100
* Jókai Mór	A kőszívű ember fiai	70
* Margaret Drabble	A vízesés	54
*	Basic Programozási Segédlet	78
* Katona József	Bánk bán	55
* Dr. Úry László	COMMODORE 64 felh. kk.	360
* Mihail Solohov	Csendes Don	34
* Móra Ferenc	Dióbél királyfi	25
Sarkadi Imre	Drámák	56
* Gárdonyi Géza	Egri csillagok	31
*	Házi mindentudó	41
* Nagy Lajos	Képtelen természetrajz	6
* Moldova György	Magyar atom	23
* James Jones	Most és mindörökké	87
*	Művészeti Lexikon	621
* Karinthy Frigyes	Napló	52
* Kertész Ákos	Névnap	33
* Moldova György	Sötét angyal	17
* Vámos Miklós	Zenga zének	46

* Josep Heller	A 22-es csapdája	92
* J. R. Tolkien	A gyűrűk ura	149
* Dumas	A három testőr	100
* Jókai Mór	A kőszívű ember fiai	70
* Margaret Drabble	A vízesés	54
*	Basic Programozási Segédlet	78
* Katona József	Bánk bán	55
* Dr. Úry László	COMMODORE 64 felh. kk.	360
* Mihail Solohov	Csendes Don	34
* Móra Ferenc	Dióbél királyfi	25
* Sarkadi Imre	Drámák	56
* Gárdonyi Géza	Egri csillagok	31
*	Házi mindentudó	41



* Joseph Heller	A 22-es csapdája	92
* J. R. Tolkien	A gyűrűk ura	149
* Jókai Mór	A kőszívű ember fiai	70
* Margaret Drabble	A vízesés	54
* Katona József	Bánk bán	55
* Mihail Solohov	Csendes Don	34
* Móra Ferenc	Dióbél királyfi	25
* Sarkadi Imre	Dramák	56

* t Joseph Heller	A 22-es csapdája	92
* t J. R. Tolkien	A gyűrűk ura	149
* t Jókai Mór	A kőszívű ember fiai	70
* t Margaret Drabble	A vízesés	54
* t Katona József	Bánk bán	55
* t Mihail Solohov	Csendes Don	34
* t Móra Ferenc	Dióbél királyfi	25
* t Sarkadi Imre	Dramák	56

Dumas	A három testőr	100
	Basic Programozási Segédlet	78
Dr. Úry László	COMMODORE 64 felh. kk.	360
Gárdonyi Géza	Egri csillagok	31
	Házi mindentudó	41
Nagy Lajos	Képtelen természetrajz	6
Moldova György	Magyar atom	23
James Jones	Most és mindörökké	87
	Művészeti Lexikon	621
Karthy Frigyes	Napló	52
Kertész Ákos	Névnap	33
Moldova György	Sötét angyal	17
Vámos Miklós	Zenga zének	46



## A MEMO KORLÁTAI

A maximális sorhossz 250.

A sor legfeljebb 16 mezőből állhat.

Egy-egy mező maximális hossza 52.

A MEMO adattárának befogadóképessége 32 Kbyte-os TV Computer esetén:

Sorhossz:	10	20	50	100	150	200	250
Sorok max. száma:	452	269	119	61	41	30	24

A MEMO adattárának befogadóképessége 64 Kbyte-os TV Computer esetén:

Sorhossz:	10	20	50	100	150	200	250
Sorok max. száma:	1639	955	424	219	148	111	89

## KÜLÖNLEGES HELYZETEK LEKEZELÉSE

Előfordulhat, hogy a program futását a CTRL-ESC segítségével meg kell szakítanunk, vagy hogy a BASIC interpreter a program futását leállítja (pl. magnószalag-hiba esetén). Ekkor kellemtelen lenne, ha a program újraindításakor az addigi munka eredménye elveszne.

Ennek elkerülésére ilyenkor a

GOTO 243<RETURN>

parancsot kiadva a MEMO letörli a képernyőt és ismét parancsra vár.



VIDEOTON

---

ELEKTRONIKAI VÁLLALAT  
SZÁMÍTÁSTECHNIKAI GYÁRA