

Magiszter

Számítástechnikai szerkesztőség

DOG

Dokumentáció
Generátor

Programvédelem nélkül!

Szoftver
IBM, PC, XT, AT
kompatibilis mikroszámítógépekre

Cod's Feed

Cape
ONS

By THOMAS Y

Photographs by

D

O

G

ELHAM

MES P. BLAIR

APHIC STAFF

Can th
Monlo

Circl
Seas

Child of catastrophe
a Bengali girl waits for
food during a famine
racked Bangladesh
autumn. Her beleaguered
tion, one of the most
densely populated on
lives in the shadow of



SCHILL—SZEGEDI

DOG

Dokumentáció Generátor

189792

SCHILL RÓBERT—SZEGEDI JÁNOS

DOG

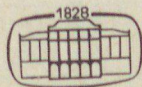
Dokumentáció Generátor

1.00 változat

MTAK



0 00002 25848 5



Akadémiai Kiadó, Budapest 1987

661001

Szoftver-dokumentáció

Sorozatszerkesztő: Pintér Tibor

A kiadványt szakmailag ellenőrizte: Meggyesházi János

©Akadémiai Kiadó és Nyomda · Schill Róbert—Szegedi János, 1987

Minden jog fenntartva

MAGYAR
TUDOMÁNYOS AKADÉMIA
KÖNYVTÁRA

M. TUD. AKADÉMIA KÖNYVTÁRA
Könyvtár 5759 / 1988

A szoftver-dokumentáció a DOG dokumentáció generátorral készült

©Akadémiai Kiadó és Nyomda · Schill—Szegedi, 1987

T a r t a l o m

1. Bevezetés.....	7
2. Általános ismertetés.....	9
3. Különleges karakterek.....	11
3.1. Kifejezéstagoló.....	11
3.2. Kifejezésösszekötő.....	12
3.3. Kifejezésválasztó.....	12
3.4. Kifejezésdeklarátor.....	13
3.5. Tabulátor.....	13
3.6. Bekezdés vége, új bekezdés megadása.....	15
3.7. Különleges kettős karakterek kifejezések.... kiemelésére	16
3.8. Ékezetes karakterek.....	18
4. A DOG parancsai.....	19
5. Szövegegységek definiálása.....	20
6. Kép megadása (\$pic).....	22
7. A különleges szövegfeldolgozások megadásának a.. vége	24
8. Táblázatok generálása.....	25
8.1. A táblázat kezdő parancsa.....	25
8.2. A táblázat fejlécének a megadása (\$shd).....	26
8.3. A táblázat jellemzőinek megadása (\$vel).....	28
8.4. A táblázat tételeinek megadása.....	31
8.5. Vízszintes vonal húzása.....	32
8.6. A többi parancs jelentése a táblázaton belül	33
9. A lap és szövegegységek formátumát leíró.....	35
parancsok	
9.1. Lapméret definiálása.....	35
9.2. A lap és szövegegység formátumának..... beállítása	36
10. Szöveg közbeni vezérlő parancsok.....	40
10.1. Lapváltás.....	40
10.2. Üres sor megadása.....	40
10.3. Tabulátorpozíciók állítása.....	41
11. Egyéb parancsok.....	42
11.1. A dokumentum címlapjának a megadása (\$tit)	42
11.2. A szabványos nevek megadása (\$sum, \$reg)...	42
11.3. Beágyazott input file megadása (\$inp).....	43
11.4. A szöveges lapok azonosítása.....	43
11.5. Élőfej nyomtatásának kérése.....	44

12. A tartalomjegyzék létrehozása és nyomtatása....	45
(\$con)	
13. Az indextételek megadása és nyomtatása (\$rg)...	46
14. Betűtípus és soremelés állítása, üzenetküldés..	47
konzolra	
14.1. Betűtípus állítása.....	47
14.2. Soremelés állítása.....	47
14.3. Feldolgozás közbeni üzenetküldés.....	47
15. A DLOOK, CONVERT és CEDIT segédprogramok.....	48
használata	
15.1. A DLOOK segédprogram.....	48
15.2. A CEDIT és CONVERT segédprogramok.....	48
15.2.1. A CEDIT program használata.....	49
15.2.2. A CONVERT program.....	51
16. A DOG használata.....	53
17. A parancsok rövid ismertetése.....	54
18. Példa.....	61

1. Bevezetés

A DOG egy RUNOFF típusú szövegformázó, amely speciális parancsokkal felkészített forrásfile alapján nyomdakész, (fénymásolásra alkalmas) formázott szöveget készít.

A DOG elsősorban a magyar nyelvű dokumentációk, levelek, formanyomtatványok, stb. feldolgozására alkalmas.

A DOG a kijelölt input file-(ok)ból egy köztes file-t készít, amely a CONVERT programmal nyomtatható.

Ellentétben a népszerű WordStar típusú szövegformázókkal a DOG nem "interaktív", az általa készített dokumentum csak teljes elkészülte után ellenőrizhető a DLOOK programmal. (A DLOOK, CONVERT és CEDIT segédprogramok használatát az utolsó fejezet ismerteti.)

Ezért a hátrányért cserébe a DOG magasszintű szolgáltatásai kárpótolnak.

A DOG lehetőségei:

- teljes magyar ékezetes karakterkészlet-kezelés, nyomtatás;
- keretezett táblázatok formázása;
- különböző mátrixnyomtató típusok lehetőségeinek kihasználása a CEDIT és CONVERT programok segítségével;
- ötszintű fejezetkezelés;
- automatikus sorszél kiegyenlítés (széthúzás);
- fejlett bekezdéskezelés;
- feltételes szóelválasztási mechanizmus;
- szövegkiemelési lehetőség aláhúzással, kövérítéssel és dőlt betűvel;
- INCLUDE mechanizmus lehetősége a dokumentáció forrásában;

- tartalomjegyzék készítése;
- index készítése;
- változtatható lap, sor, betűméret;
- alakítható lapformátum;
- lábsor vagy előfej készítése;
- fel nem készített forrás intelligens kezelése.

2. Altalános ismertetés

Ebben a részben ismertetjük a dokumentáció forrásformátumát, és annak feldolgozási módját.

Minden input sorról elmondható, hogy az alábbi két típus valamelyikének felel meg:

- A) szöveg, azaz a sor mint szövegrészlet (vagy annak folytatása) kerül feldolgozásra, ilyenkor a feldolgozás a szöveg környezetétől függ (ld. később);
- B) parancs, azaz ezt a sort mint "DOG" parancsot dolgozzuk fel és ez a sor a szöveg egy adott részének különleges feldolgozását adja meg, néhány szövegszerkesztési paramétert határoz meg vagy bizonyos értékeket definiál (pl.: tabulátorpozíciókat vagy a tartalomjegyzék nevét stb.).

A parancs típusú sor mindig "\$" karakterrel kezdődik és a parancs nevével folytatódik (pl.: \$ctr).

"Szöveg" típusú input sornak nevezzük mindazon sorokat, amelyek kezdete nem egyezik meg egyetlen paranccsal sem (pl.: Lajos, vagy \$leo). Ezeket a sorokat a környezetük függvényében különböző módokon dolgozhatjuk fel:

- A) Szabványos szöveg létrehozása. Minden olyan eset ami a B), C), D), E) esettől eltér. Tehát ezek az input sorok azt adják meg, hogy a szöveget olyan módon kell feldolgozni, mintha azt egy írógépen gépelték volna le. Ha a szövegben több sor van tagolás nélkül egymás után (azaz sem üres sor, sem parancs nincs a szövegrészletben), akkor az utolsó sort kivéve az összes sor jobbra kiigazítódik. Ez azt jelenti, hogy szükséges számú szóköz beillesztődik a szavak közé, illetve a következő sorból annyi szó kerül föl, hogy a legutolsó szó legutolsó karaktere a sor jobb szélső pozíciójára kerüljön.
- B) Szövegegység név létrehozása. (Szövegegységnek a fejezet-szinteket nevezzük.) Úgy dolgozzuk fel, hogy a szöveg-

egység ott kapjon nevet, ahol azt az input file-ban megadtuk (ez a név az input file-ban közvetlenül a parancs utáni sorban szerepel) -- lásd (5. pont).

- C) Kép nyomtatása. A szövegrészre nem történik formázás, változatlan formában kerül nyomtatásra -- lásd a képdefiniálási parancsnál (6. pont).
- D) "Középre" helyezett kép nyomtatása. Hasonló a C) esethez -- lásd a középre helyezett kép parancsnál vagy annak a definíciójánál (6. pont).
- E) Táblázattételek. A szöveg egy megadott táblázat tételeit határozza meg -- lásd a táblázat definiálási parancsnál (8. pont).

A felhasználónak lehetősége van (a később ismertetett módon) a magyar helyesírásnak megfelelő ékezetes karakterek használatára. Az input szöveg létrehozása történhet bármely szövegszerkesztő segítségével, a kézikönyvben ismertetett szabályok szerint. A létrehozandó dokumentum formátumát parancsok segítségével határozhatjuk meg. Általában egy dokumentáció különböző mélységű fejezetekre (ezeket hívjuk szövegegységnek) tagolódik. Ezeknek a mélységeknek a számozását, kezelését a DOG automatikusan elvégzi. A lap formátuma szabályozható. Egy lap alapértelmezés szerint a lapszámmal kezdődik. A lapszámozás lehet folyamatos, fejezeten belül folyamatos és el is maradhat. A sor szabályozható hosszúságú, a bal margó állítható, meghatározható a lapra kerülő sorok száma.

A lap azonosítása kétféleképpen lehetséges:

- élőfejjel, ami azt jelenti, hogy a lap tetején kiíródik egy azonosító szöveg és a lapszám, és ez vízszintes vonallal alá lesz húzva. A lapszám mindig a lap külső szélén jelenik meg, tehát a bal oldali lapokon a szöveg előtt, a jobboldaliakon pedig a szöveg után;
- lábsorral, tehát a lap alján vízszintes vonal íródik ki ez alá pedig a lap azonosító szövege.

A lap azonosító szövege páros lapon a dokumentáció címe, páratlan lapon a fejezet címe.

Ezek a szolgáltatások letilthatók.

3. Különleges karakterek

A különleges karaktereknek a következő funkcióik vannak:

- kifejezéseket definiálnak és azok feldolgozását adják meg. Ezek a karakterek kifejezések soron belüli elhelyezését is befolyásolják. Ilyen a kifejezéstagoló, a kifejezésösszekötő, a kifejezésválasztó és a kifejezésdeklarátor;
- a kifejezés pozicionálása. Két olyan karakter van, amely befolyásolja a kifejezések soron belüli kezdő pozícióját (a balra történő kiegyenlítést). Ezek a karakterek: a tabulátor és a bekezdés kezdete;
- a teljes kifejezés különleges nyomtatása (a kifejezés kihangsúlyozása).

3.1. Kifejezéstagoló

A kifejezések tagolására minden esetben a szóköz karaktert használjuk. Tetszőleges számú szóközt egyetlen szóköznek értelmezzünk. Ez a karakter egy "helyettesíthető szóközt" reprezentál, amit az output szövegben legalább egy valóságos szóközzel helyettesítünk attól függően, hogy mennyi szóköz szükséges a sor jobbra igazításához.

Megjegyzés:

Éppen ezért, ha gépelünk kövessük a szokásos gépelési stílust, a gépelésnél tehát a mondatok után legalább egy szóközt gépeljünk be és ugyanígy gépeljünk szóközt az összetettebb mondatokban a mondatrészeket határoló karakterek után is.

3.2. Kifejezésösszekötő

Ez a különleges karakter lehetővé teszi, hogy kettő (vagy több) egymás után következő szót egyetlen " oszthatatlan " kifejezésnek adjunk meg, azaz ezt a kifejezést minden esetben egy sorban kell kinyomtatni. Alapértelmezés szerint ez a legtöbb terminálon a hullámvonal karakter, ASCII kódja a &7e, de ezt meg lehet változtatni \$ctr paranccsal. Nyomatáskor ez a karakter egy valós szóközzel helyettesítődik (ezen túlmenően az így összekötött kifejezések nem szakadnak el egymástól, ha a sor jobbra kiigazítódik).

Megjegyzés:

Ezt a karaktert mindig olyan esetben használjuk, ha el szeretnénk kerülni azt, hogy az output sor vége két (vagy több) szót elválasszon egymástól, hiszen nem tudjuk, hogy az output sor ténylegesen hol végződik.

3.3. Kifejezéselválasztó

Ezzel a karakterrel megadhatjuk a kifejezés lehetséges elválasztásának a helyeit a sor végén. Standard esetben ez a backslash, ASCII kódja &5c, de ezt meg lehet változtatni a \$ctr paranccsal.

A karakter funkciója a következő: ha a karakterrel létrehozott részkifejezés az aktuális soron még elfér, akkor a kifejezés egybe fog kinyomtatódni és a kifejezéselválasztó hatástalan lesz. A kifejezés ebben az esetben teljes egészében az aktuális sorba kerül. Ha azonban a kifejezéselválasztó már nem helyezhető el az aktuális sorba, akkor csak az első részkifejezés fog az aktuális sorra kerülni az elválasztójel (-) karakterrel együtt és a második részkifejezés a következő sor elejére nyomtatódik. Pl.: ha egy sorban szerepel a számítástechnika szó de csak 10 karakter hely van a sorban, nem fér ki. Ha kifejezéselválasztóval adjuk meg, akkor az első része (számítás-) még elfér, s csak a többi kerül a másik sorba, mint 9 sorral feljebb a "kife-jezés" szónál.

Megjegyzés:

A gyakorlati tapasztalatok alapján azt mondhatjuk, hogy a legtöbb nyelvben -- ahol a szavak érezhetően hosszúak -- még az is redundáns lesz, ha a láthatóan hosszú kifejezésekre (szavakra) előre megadunk elválasztásokat. Ennél sokkal jobb stratégia az, ha a dokumentáció első generálása előtt ezeket a szavakat még nem választjuk el és a dokumentáció nyomtatott képe alapján ott adjuk meg ezeknek a szavaknak az elválasztását, ahol az egész szó a következő sorba került és így az előző sorban nagy hely maradt üresen. Látni fogjuk, hogy ez nem túl gyakran fordul elő, sőt tudnunk kell, hogy egy jól megírt dokumentáció mindig jól van szerkesztve is, és egy kifejezés elválasztása befolyásolja azt is, hogy a jól kialakított szövegszerkezet hogyan fog kinézni.

3.4. Kifejezésdeklarátor

Ezzel a deklarátorral adhatunk meg olyan kifejezéseket, amelyek tetszőleges karaktereket tartalmaznak, még a különleges karaktereket is (beleértve a szóközt). A kifejezésdeklarátor a "|" karakter ("a szaggatott függőleges vonal", ASCII kódja &7c).

Az első kifejezésdeklarátor adja meg a kezdetét egy (fel nem osztható) kifejezésnek, a második pedig a végét. Ha (rendszerint gépelési hiba miatt) a második (záró) karaktert elhagyjuk, akkor a sor vége lezárja a kifejezést. Ebből következik, hogy egy sornál nagyobb kifejezés nem adható meg egyben! Kifejezésdeklarátorát általában akkor használunk, amikor több szóközt tartalmazó kifejezéseket írunk, ilyenkor a szóközők száma gépeléskor rögzíthető.

3.5. Tabulátor

Ez a karakter lehetővé teszi, hogy olyan pozíciókat (szimbólikusan - nem számszerűen), adjunk meg, amelyekre az input szövegben hivatkozhatunk, azaz a tabulátorkarakter utáni szöveg a megadott

pozíciótól kezdődően lesz elhelyezve. A "szimbolikus" szó azt jelenti, hogy a \$tab paranccsal az input file-ban 20 pozíció beállítására van módunk és a tabulátorkarakter csak arra szolgál, hogy ebből az értékhalmból a következőt kiválasszuk. Alapértelmezés szerint ez a karakter a bal kapcsos zárójel, ASCII kódja &7b, de ezt is meg lehet változtatni a \$ctr paranccsal. A tabulátorkarakter sohasem nyomtatódik, kivéve a "kép" kezelést. Ezt a karaktert mindig úgy értelmezzük, mint kifejezéstagolót.

A tabulátor a különleges karakterekre a következő hatást fejti ki:

- a tabulátorkarakter előtt vagy után lévő kifejezés-összekötő és kifejezésválasztó karaktert nem veszi figyelembe (nem "funkcionálnak" - mivel a funkciójuknak nincs jelentése)
- ha egy tabulátorkarakter előtt olyan kifejezés van, amit egy kifejezésdeklarátorral adtunk meg, de a záró deklarátor karakter hiányzik, akkor a tabulátor zárja le a deklarációt. A \$tab parancs a lehetséges értékeket sorban egymás után adja meg. Az értékek elvileg növekvő sorrendben szerepelnek és minden tabulátorkarakter ebből az értéksorból a következő értéket választja ki, úgy ahogyan azt alább ismertetjük:
- a bekezdés vége után (azaz ha a következő input sor az első pozícióban "\$" karaktert vagy tabulátorkaraktert tartalmaz), a tabulátorkarakterrel történő pozicionálás a \$tab paranccsal elsőnek megadott értékkel kezdődik;
- ha a megadott értékek közül, már mindet kiválasztottuk (azaz az aktuális bekezdésben már volt 20 db helyesen értelmezett tabulátorkarakter -- tehát a tabulátor aktuális pozíciója a legutolsó lehetséges értéken van), akkor hibaüzenetet fogunk kapni és a tabulátor aktuális pozíciója változatlan marad;
- ha a következő érték kisebb, mint a nyomtatási sor tényleges pozíciója (azaz az előző kifejezés olyan pozíción végződik, ami a tabulátor által megjelölt pozíció után van), akkor a tabulátor aktuális pozíciója azonos lesz a következő betűpozícióval, azaz a tabulátor szóközként funkcionál.

Felhívjuk a figyelmet arra, hogy a tabulátorkarakter után következő szóközöket a sorkiegyenlítés figyelmen kívül hagyja.

- ha a tabulátor után a dokumentáció szövege több soron keresztül folytatódik (azaz a bekezdésben több sor van), akkor ez a szöveg a tabulátorpozíció után (balra igazítva) fog megjelenni. Azokat a bekezdéseket írhatjuk le így, amelyeknek nem kell a sor első pozícióján kezdődni (erre példaként szolgál a tabulátor viselkedését leíró iménti felsorolás.

3.6. Bekezdés vége, új bekezdés megadása

A szöveg típusú input sorokat (amelyek nem tartalmaznak parancsot) egymás után láncoljuk, és ebből a láncolásból áll elő egy bekezdés egészen addig, amíg ezt le nem zárja egy záró sor. Egy ilyen (záró) sor vagy a "\$" (dollár) karaktert vagy tabulátort tartalmaz az első pozícióján, vagy pedig üres. Ennek a "záró" sornak a hatásaként létrejön az előtte lévő bekezdés utolsó sora. Ez az utolsó sor nincs megnyújtva, azaz nem lesz jobbra kiigazítva.
Ime.

A "\$" vagy tabulátorkarakterrel kezdődő sorok további hatásai:

- a tabulátorkarakterrel kezdődő sornál a tabulátor pozíciója (aktuális pozíció) a legutóbbi \$tab paranccsal megadott értékek közül az elsőre állítódik be (lásd fentebb);
- a "\$" karakterrel kezdődő sor mindig lezárja az előző bekezdést a leírtak szerint, és azután kezdődik a kiértékelése (parancssor vagy üres, stb.);
- a "\$"-al kezdődő sor és ennek a sornak a kezdete nem esik egybe egyetlen paranccsal sem, így ez ismét egy szöveg típusú sor (az új bekezdés első sora). Ezen túlmenően ez a karakter befolyásolja az első kifejezés pozícióját ebben a bekezdésben: ha ezután közvetlenül egy kifejezés következik (pontosabban egy nem szóköz-karakter), akkor ez a kifejezés a bal margó+5. pozíción fog kezdődni. Megjegyezzük, hogy

ezzel a módszerrel az első kifejezést az 5. pozíciótól kezdve jobbra tolva helyezhetjük el, ha arra van szükségünk, hogy a bal margóhoz közelebb kerüljön, akkor használjuk a tabulátort;

- a "\$" karakter a szövegtípusú sorban (az előző eset) úgy szerepel, mint az előző bekezdést lezáró karakter (és meghatároz egy újat). Ez csak akkor érvényes, ha az input sor első pozícióján van a "\$". A szövegtípusú sor belsejében ezt a karaktert is egy kifejezéshez tartozó karakternek vesszük (kivéve azt az esetet, amikor ez egy különleges kettős karakter első karaktere);
- olyan input sor, amely csak a "\$" karaktert tartalmazza és azt is az első pozíciójában (azaz a sor többi része üres), vagy a sor teljesen üres. Az ilyen sort úgy értelmezzük mint \$spa parancsot, azaz hatásaként egy üres sor jön létre a dokumentációban (lásd a \$spa parancsnál);
- ha a sor a "\$" karakterrel kezdődik, a tabulátor aktuális értéke 1 lesz (azaz a sorok bal oldala az első pozíciótól kezdve lesz kiegyenlítve) és a tabulátorpozíciók kiválasztása a tabulátorkarakterrel ismét a legutóbbi \$tab parancs által megadott értékek közül az elsővel fog kezdődni;
- a fent megadott szabály alól van kivétel (máskülönben ez nem lenne szabály):
 - * az összes aláhuzást kijelölő kettős karakter
 - * a \$rg. indextétel kijelölő parancs

3.7. Különleges kettős karakterek kifejezések kiemelésére

Ezek a különleges kettős karakterek a kifejezés különleges nyomtatását (kiemelését) adják meg. Használatuk általános feltételei a következők:

- a különleges kettős karakternek csak akkor van különleges értelmezése, ha közvetlenül egy kifejezés előtt szerepel (ez kezdi a kifejezést).

- * ha egy kifejezést a kifejezésdeklatátorral adtunk meg, akkor az első deklatátor fogja jelezni az ilyen kifejezésnek a kezdetét, azaz a különleges kettős karaktert ez elő a deklarátor elé kell gépelnünk;
- * ha két kifejezés össze van kötve a kifejezés-összekötővel, akkor ezek egyetlen kifejezést alkotnak, így tehát a különleges kettős karaktert ezen két kifejezés közül az első elé kell helyeznünk (és ekkor a teljes "összekapcsolt" kifejezésre fog vonatkozni)
- ha a különleges kettős karaktert nem követi kifejezés (csak önmagában álló kifejezésként szerepel), akkor teljesen hatástalan (nem vesszük figyelembe).

A különleges kettős karakterek állandó értékre vannak előre definiálva és ez nem változtatható meg.

A kettős karakterek és jelentésük a következő:

- dollár és aláhúzás (\$ és _), kódjai &24 és &5f. A kifejezés ugyanabban a sorban lesz aláhúzva mindaddig, míg egy szóköz vagy sorvége karakter nem jön. Ez csak soron belüli aláhúzásra alkalmas. Az aláhúzás úgy történik, hogy ugyanabban a sorban (ahol az így aláhúzott kifejezés előfordul) felülnyomtattunk. Az aláhúzás a "_" karakterrel történik (aláhúzás karakter, kódja &5f);
- dollár és 2 aláhúzás (\$, _ és _), kódjai &24, &5f és &5f. A kifejezés alá lesz húzva mindaddig, amíg újra a dollár és aláhúzás, aláhúzás karaktersorozat nem következik. Ez lehet többsoros aláhúzás is. Az aláhúzás az előbb leírt módon történik.
- dollár és kettőskereszt (\$ és #), kódjai &24 és &23. A kifejezés ugyanabban a sorban lesz kövérítve mindaddig, míg egy szóköz vagy sorvége karakter nem jön. Szabályai azonosak az aláhúzás szabályaival. Kövérítés csak olyan sornyomtatón lehetséges, melynek megfelelő parancsa.
- dollár és 2 kettőskereszt (\$, # és #), kódjai &24, &23 és

&23. A kifejezés kövéritve lesz mindaddig, amíg újra a dollár és kettőskereszt, kettőskereszt karaktersorozat nem következik. Ez lehet többsoros kövérités is.

- dollár és per-jel (\$ és /), kódjai &24 és &2f. A kifejezés ugyanabban a sorban lesz dőlt beűs mindaddig, míg egy szóköz vagy sorvége karakter nem jön.
- dollár és 2 per-jel (\$, / és /), kódjai &24, &2f és &2f. A kifejezés dőlt betűs lesz mindaddig, amíg újra a dollár és per-jel, per-jel karaktersorozat nem következik. Ez lehet többsoros dőlt betűs írás is.

3.8. Ékezetes karakterek

Az ékezetes betűket úgy üjtük be, hogy a betű elé (amelyre az ékezetet akarjuk tenni) beírjuk a következő karakterek valamelyikét:

- rövid ékezetek: ":" (kettőspont) pl. ül, öl
- hosszú ékezetek: ' (apostrof) és " (macskaköröm) pl. ír, úgy, ól, és, őr.

4. A DOG parancsai

A DOG parancsai funkció szerint több részre oszthatók:

- Szövegegységek definiálása (\$cpt, \$sec, \$prg, \$art, \$itm)
- Szerkesztett és középre helyezett képparancsok (\$pic, \$mid)
- Táblázatgeneráló parancsok (\$stb, \$shd, \$vel, \$unl)
- Különleges feldolgozást lezáró parancs (\$eoc)
- Indextétel azonosítása (\$rg.)
- A szövegegységek és lapformátum leíró parancsok (\$par, \$ctr)
- Szöveg közbeni vezérlő parancsok (\$pag, \$spa, \$tab)
- Egyéb parancsok (\$inp, \$tit, \$sum, \$reg, \$con, \$eli, \$nor, \$cnd, \$dou, \$nod, \$rem, \$lhd)

5. Szövegegységek definiálása

Mint már az előzőekben utaltunk rá, szövegegységnek a fejezetszinteket nevezzük. A szövegegységek automatikusan mélység szerint sorszámozódnak.

Öt szövegegység (fejezet) szintet különböztetünk meg:

!ssz!	név	szint	számozás
! 1.!	cpt	fejezet	XX.
! 2.!	sec	alfejezet	XX.YY.
! 3.!	prg	bekezdés	XX.YY.UU.
! 4.!	art	cikkely	XX.YY.UU.WW.
! 5.!	itm	tétel	XX.YY.UU.WW.ZZ.

A szövegegységszintek definiáló parancsai:

```
$cpt/<szám>.[lapazonosító szöveg]  
$sec/<szám>.  
$prg/<szám>.  
$art/<szám>.  
$itm/<szám>.
```

ahol a szám megadja, hogy a parancsot követő hány sor tartozik az adott szövegegység címéhez. A megadott számú soron belül semmiféle parancsfigyelés, sorszéthúzás nem történik, csak az ékezetes karakterek konvertálása, tehát a sorok változatlan formában lesznek kinyomtatva. A <szám> alapértelmezése egy.

A fejezetcímek kiírásánál a program minden esetben lapot emel, ha a címet megelőző üres sorok, a címet követő üres sorok, a címsor és 5 sor a fejezetből nem fer már re az aktuális lapra. A fejezetszintet azonosító parancs hatására automatikusan létrejön a megfelelő fejezetszám.

A "\$cpt" parancsnak van egy külön paramétere, a lapazonosító szöveg. Minden lap alján vagy tetején kérhető az azonosító szöveg

kiírása. Ha ezt nem adjuk meg a "\$cpt" parancs paramétereként, akkor a "\$cpt" címét fogja a lábsorba vagy az előfejbe nyomtatni a páros lapszámú oldalakon mindaddig, míg egy új "\$cpt" parancs nem jön. (A lábsor ill. előfej nyomtatása a "\$par" parancsban letiltható. Ebben a könyvben letiltottuk, ld. a 18. pont példájának 3. sorát.)

Minden szövegegység címe automatikusan része a tartalomjegyzéknek. (Felülbíráható a "\$par" parancsban.)

Egy szövegegység kezdődhet:

- a következő páratlan lapon
- a következő páros lapon
- a következő lapon (ez az alapértelmezés az első (cpt) szintre)
- az aktuális lapon (ez az alapértelmezés az alsó 4 szintre)

Ez szintén felülbíráható a "\$par" parancsban.

6. Kép megadása (\$pic)

Ezzel a paranccsal olyan egymás után következő input, szöveg típusú sorokat adhatunk meg, ahol egy input sornak a dokumentáció egy sora felel meg és egy input sorból egy output sor keletkezik. Ez ténylegesen azt jelenti, hogy az input sorokat mindenfajta formázás nélkül nyomtatjuk ki.

A parancs szintaxisa:

`$pic[/<szám>,tc].`

A paraméterek jelentése:

- a <szám> paraméter megadja a képet alkotó sorok számát. Ez a szám ugyanakkor megadja a dokumentáció nyomtatott formájában a kép sorainak a számát is. Ennek alapértelmezés szerinti értéke &7FFF, így tehát, ha nem adjuk meg a számot, akkor a kép addig fog tartani, amíg az inputon egy "záró" parancs nem lesz (ezt lásd később);
- a tc menmonik megadása azt jelzi, hogy az ékezetes karaktereket át kell kódolni. A tc paraméter megadása nélkül a kifejezések (input sorok) abban a formában fognak kinyomtatódni, ahogy az inputon szerepelnek.

A kép létrehozásának a szabályai a következők:

- a képet alkotó input sorok nem lehetnek parancssorok, ez alól kivétel a \$inp parancs, amelynek végrehajtását a képen belül elfogadjuk (a részleteket lásd az input file-ok vezérlésénél). Ez a tulajdonság lehetővé teszi, hogy forrásprogramot másoljunk a dokumentum szövegébe.

Példa (a zárójelek között megjegyzések vannak):

```
$pic. (kép megadása)
$inp/source. (a forrásprogramot tartalmazó file)
$eoc. (a kép lezárása)
```


- a kép nyomtatása az alábbi két esetben fog befejeződni:
- * a megadott <szám>-ú szövegtípusú input sor feldolgozása befejeződött;
- * az input sorok között parancs típusú sor van (kivéve a \$inp parancsot). Ennek a módszernek a használatakor kötelező, hogy a záró parancs a \$eoc parancs legyen (ez a parancs mindig lezár néhány különleges feldolgozást. Ezt kényelmes használni, mivel a képet alkotó sorokat nem kell leszámolnunk -- elegendő, ha elég nagyra adjuk meg a <szám>-ot, vagy meghagyjuk annak az implicit értékét, és a képmegadást egy másik paranccsal lezárjuk);

A képnek egy speciális válfaja a középre helyezett kép. Megadása a \$mid paranccsal lehetséges. A képet alkotó sorokat a DOG úgy helyezi el, hogy szimmetriatengelyük a sor közepe legyen. Ezt főleg a címlapon lévő elnevezések megadásakor célszerű használnunk.

A parancs szintaxisa:

\$mid[/][inl=]<szám>,tc].

A középre helyezett kép létrehozásának paraméterei és szabályai pontosan megegyeznek azzal, amit az előzőleg a képnél elmondtunk. Csak a "kép" helyére kell mindenütt "középre helyezett kép"-et gondolnunk, a "\$pic" helyett pedig "\$mid"-et.

7. A különleges szövegfeldolgozások megadásának a vége

Ez a parancs rendszerint azt jelöli, hogy a szövegtípusú input sorok különleges feldolgozását be kell fejezni, azaz ez a parancs zár le néhány különleges feldolgozást megadó parancsot. Ezzel a paranccsal lehet lezárni a \$pic (kép), a \$mid (középre helyezett kép), a \$shd (táblázat fejléce), a \$stb (táblázat) parancsokat - lásd még a megfelelő parancsok leírását is. Ha ezt a parancsot nem a fenti szövegkörnyezetek valamelyikében használjuk, akkor nincs hatása.

A parancs szintaxisa:

\$eoc.

8. Táblázatok generálása

A DOG egyik igen kényelmes szolgáltatása a táblázatok létrehozásánál a táblázat generálása. Tágabb értelemben táblázatnak nevezzük azt a területet, amely be van keretezve és függőleges vonalakkal oszlopokra van osztva, ezen oszlopok között pedig valamilyen szöveg van. Ezt a keretezést, oszlopokra tagolást, az oszlopon belüli szöveghelyezést, fejléckekezelést lehet a DOG táblázatgeneráló parancsaival automatizálni.

Egy táblázat megadásának általános szerkezete az input file-ban:

```
$stb.                (táblázat megadása)
$shd.                (táblázat fejlécének megadása)
$vel/.....         (a fejléc függőleges vonalainak pozíciója)
                     <a fejléc szövege>
$eoc.                (a fejléc vége)
$vel/.....         (a táblázat függőleges vonalainak pozíciója)
                     <a táblázat szövege>
$eoc.                (a táblázat vége)
```

Vizsgáljuk meg az egyes parancsok pontos jelentését.

8.1. A táblázat kezdő parancsa

A parancs formája:

```
$stb/[num=< sorok>],[vch=< függőleges-vonal>],?
      [hch=< vízszintes-vonal>],[cch=< metszéspontok>],?
      tc,unl,spa,nsph].
```

A paraméterek jelentése:

- num megadja, hogy hány sor után kell a táblázatban egy vízszintes elválasztóvonalat (automatikusan) beilleszteni. Ez lehetőséget ad arra, hogy a táblázatot két részre osszuk fel. Alapértelmezés szerinti értéke &7FFF, ami gyakorlatilag azt jelenti, hogy nem kerül sor ilyen beillesztésre;

- spa az elválasztó sor egy üres sor legyen, azaz olyan sor, amelyikben nincsenek tételek, csak függőleges vonalak. Alapértelmezés szerint ez érvényes;
- unl az elválasztó sor egy vízszintes vonal legyen. A spa és unl parancsok kölcsönösen kizárják egymást, egyszerre csak az egyik szerepelhet. Ha mégis mindkettő meg van adva, az érvényes parancs az lesz, amelyik a parancssorban később van;
- vch ebből a karakterből lesz a függőleges oszlop. Alapértelmezés szerint erre a "|" -et (a függőleges szaggatott vonal) használjuk;
- hch ebből a karakterből lesz a vízszintes vonal. Alapértelmezés szerint ez a "-" (elválasztójel) karakter;
- cch a metszéspontokban használt karakter, amikor a függőleges és vízszintes vonalak metszik egymást. Alapértelmezés szerint ez a "+" (összeadásjel);
- nspv a táblázat függőleges vonalaihoz állítja a szöveget a táblázaton belüli elhelyezési mód függvényében, amelyet a \$vel parancs határoz meg (ld. 8.3 pont). Alapértelmezés szerint egy szóköz kimarad;
- nsph közvetlenül a vízszintes vonalához állítja a szöveget mivel az alapértelmezés szerint egy-egy sort kihagy;
- tc a tételek (kifejezések) átkódolásra kerülnek. Alapértelmezés szerint nem történik átkódolás.

8.2. A táblázat fejlécének a megadása (\$shd)

Ez a parancs lehetővé teszi a táblázat fejlécének, azaz a táblázat azon részének megadását, amellyel a táblázat oszlopait azonosítjuk. Ennek a fejlécnek meg kell jelennie minden olyan lapon, ahol a táblázat folytatódik (ha a táblázat hosszabb egy oldalnál).

A parancs szintaxisa:

\$shd.

A fejléc létrehozásának a szabályai:

- a fejléc elhagyható;
- egy táblázatban tetszőleges számú fejléc, tetszőleges helyen megadható;
- a fejléc megadását a \$eoc paranccsal vagy egy másik \$shd paranccsal zárhatjuk le;
- ha a táblázat több lap hosszúságú, akkor a legutoljára megadott fejléc fog automatikusan ismétlődni minden lap tetején. Ez a következő korlátozásokat eredményezi:
 - * a fejlécen belül nincs lapváltás
 - * a \$inp parancs használata a fejlécben nem megengedett,
 - * ha a fejléc függőleges vonalainak pozíciói eltérőek a táblázat többi részében lévő függőleges vonalak pozícióitól, akkor kötelező használnunk a \$vel parancsot.
- ha netán a táblázaton belül újabb fejlécet nyitunk, akkor automatikusan egy vízszintes záróvonal nyomtatódik ki.
- a fejléc definíciójának a vége (\$eoc) újabb vízszintes vonalat ír ki;
- azokban az oszlopokban, ahol nem adtunk meg semmilyen kiegyenlítést a \$vel paranccsal, ott a fejléc minden tétele "középre" kerül (lásd a \$vel parancsot);

8.3. A táblázat jellemzőinek megadása (\$vel)

Ezzel a paranccsal megadhatjuk a függőleges vonalak pozícióit és a táblázattételek elhelyezését. Minden táblázatban először ennek a parancsnak kell szerepelnie, és csak ezt követheti a tételek megadása.

A parancs formátuma:

```
$vel[/ [p1=]<p1>,[p2=]<p2>,[p3=]<p3>,[p4=]<p4>,[p5=]<p5>,[p6=]<p6>,[p7=]<p7>,[p8=]<p8>,[p9=]<p9>,[p10=]<p10>,[p11=]<p11>,[p12=]<p12>,[p13=]<p13>,[p14=]<p14>,[p15=]<p15>,[p16=]<p16>,[j1=]<j1>,[j2=]<j2>,[j3=]<j3>,[j4=]<j4>,[j5=]<j5>,[j6=]<j6>,[j7=]<j7>,[j8=]<j8>,[j9=]<j9>,[j10=]<j10>,[j11=]<j11>,[j12=]<j12>,[j13=]<j13>,[j14=]<j14>,[j15=]<j15>,allr allm alll].
```

A paraméterek jelentése:

- a P1-P16 paraméterek megadják a függőleges vonalak pozícióit (a pozíciók 1-től kezdődnek). Maximálisan 16 függőleges vonalat lehet megadni (ez pedig 15 oszlopot határoz meg).

A pozíciók értékeinek eleget kell tenniük a következő korlátozásoknak:

- * az értékeknek növekvő sorrendben kell szerepelni úgy, hogy a két egymás melletti érték által megadott pozíciók között elhelyezkedő oszlop mérete akkora legyen, hogy legalább egy nulla hosszúságú tétel elférjen benne (ami azt jelenti, hogy a következő pozíció értéke, ha nem szerepel a táblázat paraméterei között az nspv paraméter (\$stb parancs), akkor legalább hárommal nagyobb és ha az nspv-t használtuk, akkor legalább eggyel nagyobb). Az a tétel, ami egy ilyen "üres" oszlophoz tartozik nyilvánvalóan egy üres tétel kell hogy legyen (lásd később);

- * a megadott pozíciók (vízszintesen) nem lehetnek nagyobbak, mint a sor mérete;
- * legalább két pozíciót meg kell adnunk;
- a JUST alszekció J1-J15 paraméterei adják meg oszloponként a tételek elhelyezésének módját. Itt az alábbi értékeket adhatjuk meg:
 - * Jx=L (balra) - a tétel "balra-igazítódik", azaz az oszlop bal oldali függőleges vonalához (a vonal mellett egy szóköz kimarad, ha az nspv paraméter nem szerepel a \$stb parancsban). Ez az érték alapértelmezés a táblázaton belül, azaz a táblázat fejlécén kívül;
 - * Jx=M (középre) - a tétel az oszlopok között középen lesz elhelyezve. A táblázat fejlécében ez az alapértelmezés;
 - * Jx=R (jobbra) - a tétel "jobbra igazítódik", azaz az oszlop jobb oldali függőleges vonalához (most is, mint a balra igazításnál, egy szóköz kimarad);
 - * Az ALLL, ALLR, ALLM paraméterek egyszerre adják meg a kiigazítást az összes oszlopra. Alkalmazásuk nem zárja ki azt, hogy a J1-J15 paramétereket is megadjuk, azaz értelmezésük a következő: "az összes oszlop" L (balra - ALLL), R (jobbra - ALLR) vagy M (középre - ALLM) lesz kiigazítva, kivéve azokat az oszlopokat, amelyekre a kiigazítást külön megadtuk a J1-J15 paraméterekkel.

Ha egy táblázat nem tartalmaz a fejlécen kívül \$vel parancsot, akkor a függőleges vonalak pozíciója ugyanaz lesz a táblázat belsejében, mint a táblázat fejlécében. Mivel a kiigazításokat nem adtuk meg ebben az esetben (a fejlécben) a J1-J15 paraméterekkel sem pedig az ALLL, ALLLR, ALLLM paraméterekkel, így most ezek értéke L (balra) lesz (és a fejlécben, ha nem adjuk meg a kiigazítást akkor annak értelmezése M (középre)).

Hogy jobban megértsük, lássunk erre két példát:

1. Példa:

```
$stb.  
$shd.  
$vel/1,8,23/,r.  
<a fejléc tételei>  
eoc.  
<a táblázat tételei>  
eoc.
```

Az 1, 8, 23. pozícióknál húz függőlegest. Az első oszlopban a kiigazítás M (alapértelmezés szerinti), a másodikban R (a parancs által megadott helyen) lesz, a táblázatban, vagyis a fejlécen kívül pedig az alapértelmezés szerinti L érték határozza meg a szöveg állítását.

2. Példa:

```
$stb.  
$shd.  
$vel/1,8,15,18,23.  
<a fejléc tételei>  
$eoc.  
$shd.  
$vel/1,8,23/alll.  
<a fejléc tételei>  
$eoc.  
$vel/1,23/allm.  
<a táblázat tételei>  
$eoc.
```

A példa egy kétfejléces megoldást mutat be, lássuk egy kis rajzon hogy is néz ez ki.

+-----+-----+-----+	
	<=== első fejléc
+-----+-----+-----+	
	<=== második fejléc
+-----+-----+-----+	
	<=== a táblázat teste
+-----+-----+-----+	

Az első fejlécben minden tétel az alapértelmezés szerint középre van igazítva, a másodikban az összes tétel balra igazítását írja elő az "alll" paraméter, míg a táblázat testében minden középre lesz igazítva.

8.4. A táblázat tételeinek megadása

- a tételeket tabulátorkarakterek közé tesszük;
- elvileg azok a szövegtípusú input sorok, amelyek egy táblázaton belüli sort adnak meg (legtöbbször ez egy input sort jelent), annyi tabulátorkaraktert tartalmaznak, mint amennyi függőleges vonalat megadtunk a legutóbbi \$vel parancs segítségével (a példákat lásd később);
- egy táblázatbeli tételt alkotó kifejezés két egymás után következő tabulátorjel között kell legyen, ebben a kifejezésben tetszőleges karakterek szerepelhetnek (így szóköz is lehet, ami lehetővé teszi azt, hogy egy tételt a táblázat rubrikájában a megadott kiigazítástól eltérően helyezzünk el);
- szükségünk lehet arra is hogy adott oszlopokban "üres" tételeket adjunk meg, ebben az esetben ez a tétel egyszerűen kimarad a szövegtípusú input sorból (azaz a sorban ezen a helyen egymás mellett lesz két tabulátorkarakter).

Példa (ahol a "<" karakter az érvényes tabulátorkarakter):

```
$stb.  
$vel/1,10,20,30.  
<első<<harmadik<  
<első<második<  
$eoc.
```

Az első esetben nem adtunk meg tételt a második oszlopba, a második esetben pedig a táblázat harmadik oszlopára nem volt tétel megadva. Az oszlopoknak megfelelő tételek üresek maradnak;

- A program nem tördeli több sorba a hosszú tételeket, azaz egy tétele hossza legfeljebb akkora lehet, mint a számára lefoglalt oszlop!

Megjegyzés:

Ha egy tétele láthatóan több sort igényel a táblázatban (pl. a fejlécben), akkor a felhasználóra hárul, hogy több input sorban biztosítson helyet a tételnek.

- nem adhatunk meg több tételt, mint amennyi oszlop van a táblázatban. Ha ezt figyelmen kívül hagyjuk, akkor hibaüzenetet kapunk és a "többlet" tétel megadását a program nem veszi figyelembe.

8.5. Vízszintes vonal húzása

A \$unl parancs egy vízszintes sort hoz létre a táblázatban ott, ahol a parancsot kiadtuk.

A parancs formátuma:

\$unl[/<szám> spa].

A paraméterek jelentése:

- * a <szám> paraméter hatására létrejön a kért számú vízszintes sor a táblázatban és ezen sorok előtt és után egy üres sor is nyomtatódik (attól függően, hogy a \$shd parancs nsph paraméterét használtuk-e vagy sem). Ha nem adunk meg paramétert, akkor egy vízszintes sort húz;
- * az spa paraméter a vízszintes vonal fölé és alá egy üres sort irat ki. Sohasem használható a <szám> paraméterrel együtt.

1. Példa:

```
$stb/nsph.  
$shd.  
$vel/1,30,60.  
<a fejléc tételeinek megadása>  
$eoc.  
$vel/1,30,45,60.  
$unl.  
<a táblázat tételeinek a megadása>  
$eoc.
```

így a fejléc után két vízszintes vonal lesz.

8.6. A többi parancs jelentése a táblázaton belül

Elvileg minden parancsot használhatunk a táblázatdefinícióban belül, az input file-ban használható parancsok közül. Természetesen néhány használata bizonyos környezetben tiltott és néhány parancs értelmezése egy kissé eltér attól, mint ahogyan a szabványos szöveg generáláskor végrehajtódik. Most ismertetjük a kivételeket és azokonak a parancsoknak a részletes szintaxisát, amelyeknél a táblázat megadása szempontjából erre szükség van (ha a szintaxist nem ismertetjük, akkor az megtalálható az adott parancs leírásánál):

- a szövegegységek megadását szolgáló parancsok (\$cpt, \$sec, \$prg, \$art, \$itm) és a képdefiniáló parancsok (\$pic,\$mid) használata tilos a táblázaton belül;
- a \$eoc parancs megadási környezet függvényében lezárja a fejléc vagy az egész táblázat megadását. A fejléc megadásának lezárására és a táblázat lezárására mindig ezt a parancsot kell használnunk;
- a \$inp parancsot (a "beágyazott" input file megadására szolgál -- lásd az input file-ok vezérlésénél) a fejléc megadásakor tilos használni, a táblázaton belül azonban használata megengedett, és végre is hajtódik;
- a \$pag parancs (feltételes vagy feltétel nélküli lapváltás)

jelentése a szokásos, de a hatásaként létrejövő lapváltás megtöri a táblázatot, azaz lezárja a táblázatot az aktuális lapon (ha a táblázatnak legalább egy sora kinyomtatódott). A táblázat lezárása egy vízszintes vonal kiíratását és a legutóbb megadott fejléc új lapra kiíratását jelenti. A DOG a lapdobást is végrehajtja, ha az aktuális lap betelt;

- az üres sorok megadását szolgáló parancs (\$spa) értelmezése nem változik meg, csak annyi a különbség, hogy megjelennek a táblázat függőleges vonalai.

9. A lap és szövegegységek formátumát leíró parancsok

A lap méretét és formátumát, valamint a szövegegységek formátumát a "\$ctr" és "\$par" parancsok különböző paramétereivel adhatjuk meg.

9.1. Lapméret definiálása

A lap méretét a "\$ctr" paranccsal lehet kijelölni, illetve a szövegben megváltoztatni.

A parancs formája:

```
{($ctr/[lnb=<lapméret>,[cnb=<sorméret>,  
[mar=<bal margó poz.>,  
tab=<tabu\lá\tor\ka\rak\ter>,  
exc=<kifejezes osszekoto karakter>.
```

A paraméterek jelentése:

- lnb=<lapméret>

A lapon elhelyezendő sorok számát határozza meg. Ebbe beleszámít a lapszámozás sora, az utána előírás szerint kihagyandó sorok száma, a lapalji aláhúzás és az előtte kihagyandó sorok száma valamint az alsó címsor is. Ezek levonása után kapjuk meg a ténylegesen kiíródó szövegsorok számát.

- cnb=<sorméret>

Ezzel a paraméterrel határozhatjuk meg az egy sorba kerülő karakterek számát, vagyis a lap szélességét. A beállított bal margó pozíció értéke ebből levonódik, tehát a tényleges karakterszámot a sorméret-(bal margó pozíció) adja meg. Ha pl. a sorméret 70, a bal margó pozíciója 5, akkor a sorba maximum 65 karakter kerülhet.

- mar=<bal margó pozíció>

A lap bal szélső karakterpozíciójának beállítása a feladata. A lap sorainak első karaktere ebbe a pozícióba kerül.

- tab=<tabulátorkarakter>

Az a karakter, amely tabulátorkarakterként fog funkcionálni a szövegben. Alapértelmezése a nyitó kapcsos zárójel.

- exc=<kifejezésösszekötő karakter>

Az a karakter, amely kifejezésösszekötő karakterként fog funkcionálni a szövegben. Alapértelmezése a hullámos vonal.

9.2. A lap és szövegegység formátumának beállítása

A lapok és a szövegegységek címeinek formátuma adható meg a "\$par" paranccsal. Ha szöveg közben fordul elő, megváltoztatja az előzőleg definiált formátumot.

A parancs formája:

```
$par/ [pos=<a szövegegység címének elhelyezése>,  
[sbc=<üres sorok száma a szövegegység címe előtt>,  
[sac=<üres sorok száma a szövegegység címe után>,  
[for=<a cím írási módja>,  
[sbp=<a lapszámot követő üres sorok száma>,  
[sap=<a lábsort megelőző üres sorok száma>,  
[nou],[noi],[cpl],[rci],[spn],[npn],  
[pag=<szám>],  
[cpt=<szám>].,
```


A paraméterek jelentése:

- pos = a szövegegység címének elhelyezése
A következő szövegegység kezdési helyét határozza meg.
A lehetséges értékek:
 - * op: A szövegegység a következő páratlan számú lap tetején kezdődik (odd page)
 - * ep: A szövegegység a következő páros számú lap tetején kezdődik (even page)
 - * np: A szövegegység a következő lap tetején kezdődik (next page)
 - * no: A szövegegység nem kerül új lapra (no page).
- A pos paraméter egy listatípusú paraméter. Ez azt jelenti, hogy az egymás után felsorolt (max. 5) és egymástól "-" jellel elválasztott értékek rendre az adott mélységű szövegegységre vonatkoznak. Pl.: a \$par/op-op-np-ep-no. parancs hatására az első két fejezetmélység (szövegegység szint) mindig a következő páratlan lapon, a harmadik szint az aktuálisan következő lapon, a negyedik a következő páros számú lapon, az ötödik pedig az aktuális lapon fog kezdődni. Alapértelmezés szerint az első szint a következő lapon, a többi pedig az aktuális lapon kezdődik.
- sbc = üres sorok száma a szövegegység címe előtt
Minden szövegegység címe előtt ennyi üres sor fog nyomtatódni. Szintén listatípusú paraméter, tehát minden szövegegység mélységre külön megadható.
Alapértelmezése: minden szinten a cím előtt két üres sor marad ki.
- sac = üres sorok száma a szövegegység címe után
Minden szövegegység címe után ennyi üres sor fog nyomtatódni. Ez is listatípusú paraméter.
Alapértelmezése: minden szinten a cím után két üres sor marad ki.

- for = a cím írási módja
A szövegegység címének kiírási módját határozza meg. Ez is listatípusú paraméter.
Lehetséges értékei:
 - no a cím aláhúzás nélkül lesz kiírva
 - un a cím aláhúzással lesz kiírva
 Alapértelmezése: minden cím alá lesz húzva
- sbp = a lapszámot követő üres sorok száma
Alapértelmezése: két üres sor követi a lapszámozást.
- sap = a lábsort megelőző üres sorok száma
Alapértelmezése: két üres sor előzi meg a lapalji vízszintes sort.
- nou
A lapok aljára nem kell vízszintes vonalat húzni. Ez alapértelmezésben nyomtatódik.
- noi
Nem kell lábsort nyomtatni. Alapértelmezésben minden páros lap aljára a "\$tit" parancsban megadott főcím, minden páratlan lap aljára az aktuális első szövegegység szint címe nyomtatódik ki a vízszintes vonal alá.
- cpl = szám
Az a legmélyebb szövegegység szint, amelynek címe még bekerüljön a tartalomjegyzékbe. Alapértelmezésben a szövegegységek címei mind bekerülnek a tartalomjegyzékbe.
- rcl = szám
Az a legmélyebb szövegegység szint, amelynek címe még bekerüljön az indexbe. Alapértelmezésben a szövegegységek címei nem kerülnek be az indexbe.
- spn
Kulcsszavas paraméter, a lapszámozást a fejezetszám -- azon belül folyamatos lapszámról (5.27 az ötödik fejezet huszonhetedik lapja) átállítja folyamatos

lapszámozásra (113 a dokumentáció 113. lapja). Ez automatikusan letiltja a lapalji aláhúzott sort és a lábszöveget, valamint a tartalomjegyzék nyomtatását. Ez az ún. "levél típusú" lapszámozás.

- npn
Kulcsszavas paraméter, a lapszámozás letiltására szolgál.
- pag=<szám>
A lapszámozás kezdőértékét lehet megadni
- cpt=<szám>
A fejezet kezdőértékének megadására szolgál. Előfordulhat, hogy egy kész dokumentációnak egy részét meg kell változtatni. Kényelmetlen lenne, ha ebben az esetben a teljes dokumentációt újra kéne nyomtatni. Lehetőség van arra, hogy csak a megváltoztatott fejezetet nyomtassuk újra, megadva ezzel a paraméterrel a kezdő fejezetszámot.

10. Szöveg közbeni vezérlő parancsok

Az input szöveg közben kiadható szerkesztési parancsok a következők:

\$pag - lapváltás
\$spa - üres sor
\$tab - tabulátorpozíciók állítása

10.1. Lapváltás

A parancs formátuma:

\$pag/[<szám>] [ODD] [EVE]

A lapváltás lehet direkt és feltételes. Ha csak a "\$pag." van megadva, ez a feltétel nélküli lapváltást jelenti, azaz ugrás a következő lap tetejére. Ha meg van adva az ODD paraméter (\$pag/odd.), akkor a következő páratlan lapon, ha az EVE paraméter, akkor a következő páros lapon folytatódik a nyomtatás. Ha a <szám> paraméter van megadva, csak akkor történik lapemelés, ha a <szám> darab sor már ezen a lapon nem fér el.

A három paraméter kölcsönösen kizárja egymást, mindig csak az egyik szerepelhet.

A lapemelés hatására kiíródik a lábsor, és a következő lap tetején az aktuális lapsorszám.

10.2. Üres sor megadása

Az üres sort három módon lehet megadni.

- a dokumentáció szövegében gépeléskor üres sorokat hagyunk;
- az input sor csak egy "\$" karaktert tartalmaz;
- \$spa/<szám> paranccsal.
Ez a parancs lehetőséget ad arra, hogy egy vagy több üres

sort adjunk meg a dokumentum szövegében. A <szám> paraméter adja meg az üres sorok számát. Ha nincs megadva, az alapértelmezés 1.

Az első két módon megadott üres sor -- ha nem fér el az aktuális lapon -- lapváltás után az új lapon nyomtatódik.

Ha a "\$spa" paranccsal kértük az üres sor nyomtatását, akkor a felesleges üres sorok az új lapon nem nyomtatódnak ki. Pl.: a "\$spa/5." paranccsal 5 üres sort kértünk, de a lapon már csak 3 sor fér el, a lapemelés után nem nyomtatódik ki a 2 hiányzó sor. Viszont ha 5 üres sort gépeltünk, vagy 5 csak "\$"-t tartalmazó sort, akkor a fennmaradó 2 sor a következő lapon üres sorként megjelenik.

10.3. Tabulátorpozíciók állítása

A DOG összesen 20 tabulátorpozíciót tud értelmezni.

Alapértelmezésben az:

1, 4, 8, 12, 16, 68, 72, 76 sorozat érvényes.

A tabulátorpozíciók az input szövegben bármikor átdefiniálhatók a "\$tab" paranccsal.

A parancs formátuma:

\$tab/x,x,x,x, ,x,x,x.

Ahol x az a pozíció ahová a tabulátort szeretnénk állítani. A nem állított tabulátorpozíciók értéke 0. A tabulátorpozíciók mindig a bal margótól számítódnak. Ha a tabulátorállítás kilóg a sorból, a DOG hibajelzést ír ki, és a pozíciót figyelmen kívül hagyja.

A tabulátorugrás jele az input szövegben a nyitó kapcsos zárójel. A tényleges tabulátorkarakterek csak a "kép" parancsokon belül használhatók, egyébként szóközzel helyettesítődnek.

11. Egyéb parancsok.

11.1. A dokumentum címloldalának a megadása (\$tit)

Ezzel a paranccsal lehet a dokumentáció címloldalát megadni. A címloldal a tényleges szöveg előtt szerepel, tehát nincs lapszámozása (a lapszámozás a címloldal vége után kezdődik el).

A parancs szintaxisa:

`$tit.[<a dokumentáció neve>]`

A címlap ilyen módon tehát lapszámozás nélküli szövegrész.

A \$tit parancs a dokumentáció nevét is megadja. Ezt a nevet használjuk arra, hogy a páros (bal oldali) lapokat azonosítsuk - lásd a lapszámozással foglalkozó fejezetet. Ezt a nevet a \$tit parancsban lévő "." karakter után adjuk meg. Ha a parancs szerepel a forrásban, akkor meg kell, hogy előzze az első kinyomtatandó szövegrészt.

11.2. A szabványos nevek megadása (\$sum, \$reg)

Ezek a parancsok lehetővé teszik, hogy a dokumentáció állandó részeinek elnevezéseit megadjuk.

A név megadása ténylegesen a parancs után történik (a parancs zárókaraktere a "." után). A tartalomjegyzék és index nevét a dokumentum megfelelő részében fejlécfeliratként használjuk, de azonosító szöveggént is kinyomtatjuk a megfelelő lapok aljára vagy tetejére (ld. lapozás részt).

A parancsok szintaxisa:

`$sum[/<a-lapok-számozása>].<a-tartalomjegyzék-neve>`

\$reg[/<a-lapok-számozása>].<az-index-neve>

A "\$sum" a tartalomjegyzék nevét adja meg, a "\$rem" pedig az indexét. A \$rem parancs egyúttal előírja az index létrehozását és nyomtatását is. Az első paraméter egy betű, amit a lapok számozásánál használunk (ld. a lapozásnál). Alapértelmezés szerinti értéke tartalomjegyzéknél a "T" betű, indexnél az "I". A második paraméter az index illetve a tartalomjegyzék címe. Az indexnél alapértelmezésben INDEX, a tartalomjegyzéknél pedig TARTALOM.

11.3. Beágyazott input file megadása (\$inp)

Ezzel a paranccsal másolható egy adott input file-ba egy másik file.

A parancs formátuma:

\$inp/<file-név>

ahol a <file-név> az MS-DOS által megengedett névkonvencióknak kell, hogy megfeleljen.

11.4. A szöveges lapok azonosítása

A dokumentáció minden számozott lapja tartalmazhat olyan szöveget is a lapokon, ami a lapot azonosítja. Formájának és használatának szabályai a következők:

- az azonosító szöveg nyomtatható:
 - * a lap aljára, azaz az adott lapon lévő szöveg alá (lábsor);
 - * a lap tetejére, azaz az adott lapon lévő szöveg fölé (élőfej);
- a kinyomtatás letiltható a "\$par" parancs "noi" paraméterének segítségével;

- a szöveg kinyomtatása előtt üres sorok maradnak ki, ezeknek a száma megadható a "sap" paraméterrel a \$par parancsban (alapértelmezés szerint 1 üres sor marad ki);
- a lapot azonosító szöveg előtt vagy után egy "-" karakterekből álló vízszintes vonal is kinyomtatásra kerül, de ha feleslegesnek találjuk, akkor ez megszüntethető a "nou" paraméter megadásával az említett helyen;
- az azonosító szövegre vonatkozó szabályok szerint azonosításra a következő szövegek (nevek) valamelyike használható:
 - * a "\$tit" paranccsal legutóbb megadott dokumentációnév a dokumentáció minden részénél a bal oldali, páros lapok aljára fog kerülni;
 - * a jobb oldali, páratlan lapok aljára a fejezet címe fog kerülni, ez attól függ, hogy a dokumentum mely részéről van szó:
 - . a szöveges résznél ez a legutóbb megadott fejezet neve lesz, azaz a "\$cpt" parancsnál megadott szöveg, vagy a fejezet nevének első sora, feltéve, hogy ez a szöveg nem lett átdefiniálva a "\$cpt" parancsban a "." karakter után;
 - . a tartalomjegyzék esetén ez a megfelelő rész neve, amit a legutóbbi "\$sum" (tartalomjegyzék) parancs segítségével adtunk meg (ha ilyen parancsot nem használtunk, akkor a standard szöveg lesz érvényes -- T A R T A L O M).

11.5. Előfej nyomtatásának kérése

A lapazonosítási lehetőségek egyike. Használata kizárja a lábsor nyomtatását. Az előfej azt jelenti, hogy a lap tetején kiíródik egy azonosító szöveg és a lapszám. Ez vízszintes vonallal alá lesz húzva. A lapszám mindig a lap külső szélén jelenik meg, tehát a bal oldali lapokon a szöveg előtt, a jobboldaliakon pedig a szöveg után.

A parancs formája: \$LHD.

12. A tartalomjegyzék létrehozása és nyomtatása (\$con)

A tartalomjegyzék olyan szövegegységnevek halmaza, amelyeket a "\$cpt", "\$sec", "\$prg", "\$art" és "\$itm" parancsokkal adtunk meg. A tartalomjegyzék létrehozásának és nyomtatásának a következő igen egyszerű szabályai vannak:

- a "cpl" paraméter ("par" parancs) korlátozhatja a szövegegységnevek tartalomjegyzékbe vételét - a szövegegységnevek az 1 szintű fejezettől kezdve a megadott szintig kerülnek a tartalomjegyzékbe (ennek implicit értéke 5, ekkor az összes szövegegységnév belekerül a tartalomjegyzékbe);
- csak azok a szövegegységnevek kerülnek a tartalomjegyzékbe, amelyek a dokumentáció szövegében előfordulnak;
- a következő nyilvánvaló tény az, hogy a szövegegységek a megadásuk sorrendjében rögzítődnek, és a tartalomjegyzék ilyen módon a lapszámok szerint rendezett lesz;
- a tartalomjegyzék létrehozása (azaz a tételek - szövegegységnevek bejegyzése a tartalomjegyzékbe) minden esetben megtörténik (még akkor is, ha a "\$con" funkciót nem kértük). A "\$con" parancs tehát mindössze a tartalomjegyzék nyomtatását kéri. Ha a parancsot nem adjuk meg, a tartalomjegyzék nem íródik ki;
- a szövegegységnév formátuma ugyanaz, mint ahogy a szövegben szerepel (még a szövegegységszám eltolása is megegyezik), az automatikus aláhúzás természetesen nem fog megjelenni;
- a \$cpt-el megadott szövegegység nevet az azt megelőző szövegegységnévtől egy üres sor választja el;
- a név utolsó kifejezését egy szóköz követi és ezután pontozott vonal kerül (". " karakterekből - pontokból), ami egészen a lap jobb oldaláig tart (elmaradhat, ha hosszú a szövegegység neve), ezt követi az a lapszám, ahol a szövegegység megtalálható. A lapszámok a lap jobb szélére igazítódnak.

13. Az indextételek megadása és nyomtatása (\$rg)

Az indextételek azonosítása kétféleképpen történhet:

- olyan szövegegységnevek kerülhetnek be, amelyeket a "\$cpt", "\$sec", "\$prg", "\$art" és "\$itm" parancsokkal adtunk meg. Az "rcl" paraméter ("par" parancs) előírhatja a szövegegységnevek indexbe vételét -- a szövegegységnevek alapértelmezésben nem kerülnek be az indexbe.
- a \$rg. paranccsal azonosított sorok tartalma minden esetben indextétel.
A parancs formája:

\$rg.<szöveg>

ahol a <szöveg> paraméter az indextétel szövege. Ez a parancs nem szakítja meg az input folyamatos feldolgozását, bár "\$" karakterrel kezdődik, a sorok tördelésénél nem funkcionál parancsként.

Az index létrehozása (azaz a tételek bejegyzése az indexbe) csak abban az esetben történik meg, ha a "\$reg" funkciót kértük. Ez egyben előírja a nyomtatást is.

Az indextételek a következő formában nyomtatódnak:

- névsorba rendezetten (a névelők -- a, az -- figyelembe vétele nélkül;
- a tétel mögött az oldalszámok, ahol a tételek azonosítása történt;
- azonos tételek egyszer nyomtatódnak, több oldalszámmal.

14. Betűtípus és soremelés állítása, üzenetküldés konzolra

14.1. Betűtípus állítása

Az input sorban megadott \$eli parancs "elite size" módra, a \$nor parancs "pica size" (normál) módra, a \$cnd condensed (összehúzott) módra állítja a sornyomtatót. Ez a mód marad mindaddig, amíg át nem állítjuk más módra.

14.2. Soremelés állítása

Lehetőség van minden egyes soremelés dupla soremeléssé alakítására a \$dou paranccsal. Mindaddig, amíg a \$nod parancsot ki nem adjuk, minden sor után egy üres sor nyomtatódik.

14.3. Feldolgozás közbeni üzenetküldés

A szöveg feldolgozása közben lehetőségünk van üzenet küldésére.
A parancs formája:

\$rem/pause.<szöveg>

A parancs a <szöveg>-ben megadott karaktersorozatot a képernyőre nyomtatja. Ha használtuk a "pause" paramétert, akkor a feldolgozás mindaddig nem folytatódik, míg a <szöveg> megjelenését követően egy billentyűt le nem ütünk. Ha nem használtuk, csak a <szöveg> íródik ki, a feldolgozás nem szakad meg.

15. A DLOOK, CONVERT és CEDIT segédprogramok használata

15.1. A DLOOK segédprogram

A DLOOK segédprogram lehetőséget ad a megszerkesztett dokumentáció nyomtatás előtti megtekintésére a képernyőn. A DOG (ha nem adunk meg output file-t) létrehoz egy <input file>.DTF nevű köztes file-t, amelyet inputként megadva a DLOOK programnak, a szöveg szerkesztett formában megjelenik a képernyőn. Különböző színekkel jelöli a kifejezés kiemeléseket. Piros, ha aláhúzott, kék, ha dőlt betűs, intenzív, ha kövérített a kifejezés. A DLOOK help-je a "?" megnyomására aktivizálódik. A file beolvasása után a kezelés legegyszerűbben a kurzormozgató billentyűkkel történhet.

Indítása:

```
>DLOOK <inputfile>
```

15.2. A CEDIT és CONVERT segédprogramok

Ez a programpáros lehetőséget nyújt arra, hogy a DOG által előállított köztes file-ból nyomtatható file-t állítsunk elő. Erre azért van szükség, hogy bármely nyomtató használható legyen a dokumentáció kiíratására. A CEDIT programmal elő kell állítani az átkódoló file-t, melynek felhasználásával a CONVERT program elvégzi a file átkódolását. Mindkét program konvertáló file-okkal dolgozik. A konvertáló file-okat célszerű egy könyvtárban tartani. Így viszont kényelmetlen, hogy híváskor mindig a teljes (könyvtárnévvel megadott) file-nevet kell beütni. Ezért a programok először az aktuális könyvtárban, majd ha ott nem találták a rendszer SET parancsával megadott DOGLIB könyvtárban keresik a hívott konvertáló file-t. A SET parancsot érdemes az AUTOEXEC.BAT file-ba beletenni a következő formában:

```
set DOGLIB=<könyvtárnév>
```

ahol a <könyvtárnév> nevű könyvtárban vannak a konvertáló file-ok.

15.2.1. A CEDIT program használata

A program feladata egy olyan konvertáló file létrehozása, melynek használatával a CONVERT program az átalakításokat elvégezheti. Tehát itt a CEDIT programban kell megadnunk, hogy a DOG outputfile-jában lévő ékezetes és vezérlő karakterek milyen karakternek felelnek meg a nyomtatón.

A program két képernyővel dolgozik:

- az első képernyőn adhatjuk meg a karakterkészlet egy elemének cseréjét egy másik elemre.
- a második képernyőn jelölhetjük ki bármely karaktersorozat bármilyen másik karaktersorozatra való cseréjét.

A program elindítható input file megadásával és anélkül. Az első esetben egy már meglévő átkódoló file-t akarunk javítani, a második esetben pedig egy újat hozunk létre.

A következő táblázatban példaként leírjuk, hogy melyik karaktereket mire kell cserélni a két mátrixnyomtató és a képernyő esetén.

Az első oszlopban megadjuk a betűt, a másodikban a DOG köztesfile-jában levő értéket, a harmadikban a képernyőre íratásnál használható értéket decimálisan. A negyedik oszlop a VIDEOTON gyártmányú nyomtatók, az ötödik pedig az EPSON RX típusú nyomtatók kódjait tartalmazza.

Betűk	DOG	Consol	VT21200	Epson RX
'a	130	160	224	ESC, "R", 6, "ç", ESC, "R", 0
'A	131	143	192	"A", 8, ""
'e	132	130	225	ESC "R", 2, "ç", ESC, "R", 0
'E	133	144	193	ESC "R", 5, "@", ESC, "R", 0
'i	134	161	226	"i", 8, ""
'I	135	73	73	"I", 8, ""
'o	136	162	227	ESC "R", 6, "i", ESC, "R", 0
'O	137	162	195	"O", 8, ""
:o	138	148	228	ESC "R", 5, "i", ESC, "R", 0
:O	139	153	196	ESC "R", 5, "\", ESC, "R", 0
"o	140	148	229	"o", 8, ""
"O	141	153	197	"O", 8, ""
'u	142	163	230	"u", 8, ""
'U	143	163	198	"U", 8, ""
:u	144	129	231	ESC "R", 2, "j", ESC, "R", 0
:U	145	154	199	ESC "R", 2, "J", ESC, "R", 0
"u	146	129	232	"u" 8 ""
"U	147	154	200	"U" 8 ""

Kódok	DOG	Consol	VT21200	Epson RX
aláhúzás be	150		ESC"U"	ESC "-" 1
aláhúzás ki	151		ESC"U"	ESC "-" 0
dupla nyomt. be	152		ESC"R"	ESC "G"
dupla nyomt. ki	153		ESC"R"	ESC "H"
normál betűtípus	154		ESC"-"	ESC "P" 18
elite betűtípus	155		ESC"*"	18 ESC "M"
cond. betűtípus	156		ESC"+"	ESC "P" 15
italic mód be	157		nincs	ESC 4
italic mód ki	158		nincs	ESC 5
inicializáló kar.	159			

Az inicializáló karakter arra szolgál, hogy a nyomtatónak minden lehetőségét kihasználhassuk. Ezt a karaktert a DOG induláskor azonnal kiteszi a köztes file-ba. Ha a sornyomtatónak vezérlő karaktereket akarunk adni, ezt a karaktert cseréljük a konvertálás során a vezérlő sorozatra. Ha nem akarjuk felhasználni semmire, a konvertáló táblában egy nulla hosszúságú stringre váltjuk ki.

15.2.2. A CONVERT program

Ezzel a programmal végezhetjük el az output file végleges formára hozását. Minden karaktert vagy karaktersorozatot a CEDIT programmal előállított konverter file alapján a megadottra cserél.

A CONVERT program hívása:

```
convert input_file output_file konverter_file
```


Az input file, melyet a DOG programmal állítottunk elő, alapértelmezett kiterjesztése ".DTF". Az output file-t a felhasználó köteles megadni, a rendszer file-név konvencióinak megfelelően. Ha a file-típust nem adja meg, akkor az alapértelmezés az "input_file.DOC" file-név. A konverter file-t a CEDIT segédprogrammal kell előállítani, alapértelmezett névkiterjesztése ".CNV".

16. A DOG használata

A DOG egy szövegfile-t dolgoz fel. Ezt a file-t előállíthatjuk bármilyen editorral, amelyik nem tesz speciális karaktereket a file-ba, mi a szerkesztés folyamán a NORTON editort és a Perfect Writer-t használtuk a szövegek begépelésére. A szöveg elkészülte után meg kell hívni a DOG-ot a következő módon:

>DOG input [opciók]

Opciók:

-o <fnév> : az output file

-c <fnév> : a konvertáló file neve. Ha nem adunk meg konvertáló file-t, akkor a CONVERT segédprogram nem hívódik meg, s az output file megtekinthető a DLOOK programmal. A "-c" opció használata csak teljesen kész dokumentációnál célszerű. Ha nem adjuk meg paraméterként, a CONVERT önállóan is meghívható a DOG output file-jára.

-t<betű> a karakter típusa
ha a betű:

e	-	Elite size
n	-	normál
c	-	condensed

Alapértelmezett input névkiterjesztés: .TXT

Alapértelmezett output: <inputfile>.DTF

Alapértelmezett karaktertípus: normál

Ha a DOG-ot nem -c opcióval hívtuk meg, akkor ezután a CONVERT Programot kell ráengedni az előállított file-ra.

17. A parancsok rövid ismertetése

Minden parancs alatt megadjuk azt az angol szót, melyből a rövidítés született.

```
$art[/<szám>].  
art -> article
```

Negyedik szintű szövegegység.

```
$con.  
con -> content
```

A tartalomjegyzék nyomtatásának kérése.

```
$cnd.  
cnd -> condensed
```

Condensed betűtípusú nyomtatás.

```
$cpt[/<szám>]. [<szöveg>]  
cpt -> chapter
```

Első szintű szövegegység.

```
$ctr [/lnb=<lapméret>,[cnb=<sorméret>,[mar=<bal margó poz.>,  
tab=<tabulátorkarakter>], exc=<kifejezes osszekoto karakter>].  
ctr -> control
```

Lapméret definiálása.

- lnb=<lapméret>
lnb -> line number
a lapon elhelyezendő sorok száma
- cnb=<sorméret>
cnb -> character number
az egy sorba kerülő karakterek száma

- mar=<bal margó pozíció>
mar -> margin
a lap bal szélső karakterpozíciójának beállítása
- tab=<tabulátorkarakter>
tab -> tabulator
- exc=<kifejezésösszekötő karakter>
exc -> expression continue

\$dou.
dou -> double

Dupla soremelés bekapcsolása.

\$eli.
eli -> elite

Elite betűtípusú nyomtatás.

\$eoc.
eoc -> end of command

Különleges feldolgozást lezáró parancs.

\$inp/<file-név>.
inp -> input

Beágyazott file kijelölése.

\$itm[/<szám>].
itm -> item

Ötödik szintű szövegegység.

\$mid[/tc].
mid -> middle

Középre helyezendő kép kezdete.

\$nod.
nod -> no double

Dupla soremelés kikapcsolása.

\$nor.

nor -> normal

Normál (Pica size) betűtípusú nyomtatás.

\$pic[/tc].

pic -> picture

Kép kezdetét definiáló parancs.

\$prg[/<szám>].

prg -> paragraph

Harmadik szintű szövegegység.

\$pag[/<szám> ODD EVE].

pag -> page

Lapváltás.

\$par / [pos=] <a szövegegység címének elhelyezése>,
[sbc=] <üres sorok száma a szövegegység címe előtt>,
[sac=] <üres sorok száma a szövegegység címe után>,
[for=] <a cím írási módja>,
[sbp=] <a lapszámot követő üres sorok száma>,
[sap=] <a lábsort megelőző üres sorok száma>,
[nou], [noi], [cpl], [rcl], [spn], [npn], [pag=<szám>],
[cpt=<szám>].

par -> parameters

A lap és szövegegység formátumának beállítása.

- pos = a szövegegység címének elhelyezése
pos -> positions
A lehetséges értékek:

- * op: A szövegegység a következő páratlan számú lap tetején kezdődik (odd page).
- * ep: A szövegegység a következő páros számú lap tetején kezdődik (even page).

- * np: A szövegegység a következő lap tetején kezdődik (next page).
- * no: A szövegegység nem kerül új lapra (no page).
- sbc = üres sorok száma a szövegegység címe előtt
sbc -> space before chapter
- sac = üres sorok száma a szövegegység címe után
sac -> space after chapter
- for = a cím írási módja
for -> format
Lehetséges értékei:
 - no a cím aláhúzás nélkül lesz kiírva (not)
 - un a cím aláhúzva lesz kiírva (underline)
- sbp = a lapszámot követő üres sorok száma
sbp -> space before page
- sap = a lábsort megelőző üres sorok száma
sap -> space after page
- nou = a lapok aljára vagy tetejére nem kell vízszintes vonalat húzni
nou -> no underline
- noi = nem kell lábsort vagy előfejet nyomtatni
noi -> no identify
- cpl = tartalomjegyzékbe kerülő legkisebb szövegegység mélysége
cpl -> chapter level
- rcl = indexbe kerülő legkisebb szövegegység mélysége
rcl -> register count level
- spn = folyamatos lapszámozás
spn -> single page number

- npn = lapszámozás letiltása
no page number
- pag = lapszámozás kezdő értéke
pag -> page
- cpt = fejezetszám kezdő értéke
cpt -> chapter

\$reg/rom c.[<szöveg>]
reg -> register

Az index lapszámozása.

\$rem/pause.
rem -> remark

Menetközbeni üzenetküldés a képernyőre.

\$rg.
rg -> register

Indextétel azonosítása.

\$spa[/<szám>].
spa -> space

Üres sor megadása.

\$sec[/<szám>].
sec -> section

Második szintű szövegegység.

\$shd.
shd -> set header

A táblázat fejlécének a megadása.


```
$stb[/ [num=]<sorok>,[vch=]<függőleges-vonal>?,
[hch=]<vízszintes-vonal>,[cch=]<metszésponatok>?,
tc,unl,spa,nspv,nsph].
stb -> set table
```

A táblázat kezdő parancsa.

- num hogy hány sor után kell a táblázatban egy vízszintes elválasztóvonalat automatikusan beilleszteni (number).
- spa az elválasztó sor egy üres sor legyen (space).
- unl az elválasztó sor egy vízszintes vonal legyen (underline).
- vch a függőleges vonal karaktere (vertical character).
- hch a vízszintes vonal karaktere (horizontal character).
- cch a metszésponatokban használt karakter (cross character).
- nspv a táblázat függőleges vonalai előtt és után közvetlenül ne maradjon ki üres hely (no space vertical).
- nsph a vízszintes vonalak elé vagy után ne kerüljön üres sor (no space horizontal).
- tc a tételek (kifejezések) átkódolásra kerülnek. (text convert).

```
$sum/rom c.[<szöveg>]
sum -> summary
```

A tartalomjegyzék lapszámozása.

\$unl[<szám> spa].
unl -> underline

Vízszintes vonal húzása a táblázatban.

\$vel/<függőleges vonalpozíciók>/<tételek elhelyezése>.
vel -> vertical lines

A függőleges vonalak pozíciójának és a táblázattételek elhelyezésének megadása.

\$tab/x,x,x,x,,x,x,x.
tab -> tabulator

Tabulátorpozíciók állítása.

\$tit.<szöveg>
tit -> title

Fejléc vagy lábsor kijelölése.

18. Példa

Ez a fejezet tartalmazza a könyv forrását DOG inputként.

```
$eli.  
$tit.Felhaszn'al'oi K'ezik:onyv  
$par/nou,noi,spn.  
$ctr/lmb=48,cnb=71,mar=5.  
$sum/rom.  
$con.  
$spa/15.  
$mid/tc.  
DOG
```

Dokumentum Gener'ator

Felhaszn'al'oi k'ezik:onyv

1.00 v'altozat

\$eoc.

\$spa/10.

\$mid/tc.

(c) 1987 Schill R., Szegedi J., Akad'emiai Kiad'o

\$eoc.

\$cpt.

Bevezet'es

A DOG egy RUNOFF t'ipus'u sz:ovegform'az'o, amely speci'alis parancsokkal felk'esz'itett forr'asfile alapj'an nyomdak'esz, (f'enym'asol'asra alkalmas) form'azott sz:oveget k'esz'it.

A DOG els"osorban a magyar nyelv"u dokument'aci'ok, levelek, formanyomtatv'anyok, stb. feldolgoz'as'ara alkalmas.

A DOG a kijel:olt input file-(ok)b'ol egy k:oztes file-t k'esz'it, amely a CONVERT programmal nyomtathat'o.

Ellent'etben a n'epszer"u WordStar t'ipus'u
sz:ovegform'az'okkal
a DOG nem "interakt'iv", az 'altala k'esz'itett dokumentum
csak teljes elk'esz:ulte ut'an ellen"orizhet"o a DLOOK
programmal. (A DLOOK, CONVERT 'es CEDIT seg'edprogramok
haszn'alat'at az utols'o fejezet ismerteti.)

Ez'ert a h'atr'any'ert cser'ebe a DOG
magasszint"u szolg'altat'asai
k'arp'otolnak.

A DOG lehet"os'egei:

{-(teljes magyar 'ekezetes karakterk'eszlet-kezel'es,
nyomtat'as;

{-(keretezett t'abl'azatok form'az'asa;

{-(k:ul:onb:oz"o m'atrixnyomtat'o t'ipusok lehet"os'egeinek
kihaszn'al'asa a CEDIT 'es CONVERT programok seg'its'eg'evel;

{-(otszint"u fejezetkezel'es;

{-(automatikus sorsz'el kiegyenl'it'es (sz'eth'uz'as);

{-(fejlett bekezd'eskezel'es;

{-(felt'eteles sz'oelv'alaszt'asi mechanizmus;

{-(sz:ovegkiemel'esi lehet"os'eg al'ah'uz'assal,
k:ov'er'it'essel 'es d"olt bet"uvel;

{-(INCLUDE mechanizmus lehet"os'ege a dokument'aci'o
forr'as'aban;

{-(tartalomjegyz'ek k'esz'it'ese;

{-(index k'esz'it'ese;

{-(v'altoztathat'o lap, sor, bet"um'eret;

{-(alak'ithat'o lapform'atum;

(- (l'absor vagy 'el"ofej k'esz'it'ese;

(- (fel nem k'esz'itett forr'as intelligens kezel'ese.
\$cpt.

'Altal'anos ismertet'es

Ebben a r'eszben ismertetjuk a dokument'aci'o
forr'asform'atum'at,
'es annak feldolgoz'asi m'odj'at.

Minden input sorr'ol elmondhat'o, hogy az al'abbi k'et t'ipus
valamelyik'enek
felel meg:

A) ((sz:oveg, azaz a sor mint sz:ovegr'eszlet (vagy annak
folytat'asa)
ker:ul feldolgoz'asra, ilyenkor a feldolgoz'as a sz:oveg
k:ornyezet'et"ol f:ugg
(ld. k'es"obb);

(B) ((parancs, azaz ezt a sort mint "DOG" parancsot dolgozzuk
fel 'es ez a
sor a sz:oveg egy adott r'esz'enek k:ul:onleges feldolgoz'as'at
adja meg,
n'eh'any sz:oveg\szerkeszt'esi param'etert hat'aroz meg vagy
bizonyos
'ert'ekeket defini'al (pl.: tabul'atorpoz'ici'okat vagy a
tartalomjegyz'ek
nev'et stb.).

A parancs t'ipus'u sor mindig "\$" karakterrel kezd'odik 'es a
parancs nev'el folytat'odik (pl.: \$ctr).

"Sz:oveg" t'ipus'u input sornak nevezzuk mindazon sorokat,
amelyek
kezdetek nem egyezik meg egyetlen paranccsal sem (pl.: Lajos,
vagy \$leo).
Ezeket a sorokat a k:ornyezet:uk f:uggv'eny'eben k:ul:onb:oz'o
m'odokon
dolgozhatjuk fel:

A)((Szabv'anyos sz:oveg l'etrehoz'asa. Minden olyan eset ami a B), C), D), E)

esett"ol elt'er. Teh'at ezek az input sorok azt adj'ak meg, hogy a sz:oveget olyan m'odon kell feldolgozni, mintha azt egy 'ir'og'epen g'epelt'ek volna le.

Ha a sz:ovegben t:obb sor van tagol'as n'elk:ul egym'as ut'an (azaz sem :ures sor, sem parancs nincs a sz:ovegr'eszletben), akkor az utols'o sort kiv'eve az :osszes sor jobbra kiigaz'it'odik. Ez azt jelenti, hogy sz:uks'eges sz'am'u sz'ok:oz beilleszt"odik a szavak k:oz'e, illetve a k:ovetkez"o sorb'ol

annyi sz'o ker:ul f:ol, hogy a legutols'o sz'o legutols'o karaktere a sor jobb sz'els"o poz'ici'oj'ara ker:uljon.

(B)((Sz:ovegegys'eg n'ev l'etrehoz'asa.

(Sz:ovegegys'egnek a fe\je\zet\szin\te\ket nevezze:uk.)

'Ugy dolgozzuk fel,

hogy a sz:oveg\egys'eg ott kapjon nevet, ahol azt az input file-ban

megadtuk (ez a n'ev az input file-ban k:ozvetlen:ul a parancs ut'ani

sorban szerepel) -- l'asd (5. pont).

(C)((K'ep nyomtat'asa. A sz:ovegr'eszre nem t:ort'enik form'az'as,

v'altoztatlan form'aban ker:ul nyomtat'asra -- l'asd a k'epdefini'al'asi parancsn'al (6. pont).

(D)(("K:oz'epr" helyezett k'ep nyomtat'asa. Hasonl'o a C) esethez -- l'asd a

k:oz'epr helyezett k'ep parancsn'al vagy annak a defin'ici'oj'an'al (6. pont).

(E)((T'abl'azatt'etelek. A sz:oveg egy megadott t'abl'azat t'eteleit

hat'arozza meg -- l'asd a t'abl'azat defini'al'asi parancsn'al (8. pont).

A felhaszn'al'onak lehet"os'ege van (a k'es"obb ismertetett

m'odon) a magyar helyes'ir'asnak megfelel'o 'ekezetes karakterek haszn'alat'ara.

Az input sz:oveg l'etrehoz'asa t:ort'enhet b'armely sz:ovegszerkeszt'o seg'its'eg'evel, a k'ezik:onyvben ismertetett szab'alyok szerint.

A l'etrehozand'o dokumentum form'atum'at parancsok seg'its'eg'evel hat'arozhatjuk meg.

'Altal'aban egy dokument'aci'o k:ul:onb:oz'o m'elys'eg'u fejezetekre (ezeket h'ivjuk sz:oveg\egys'egnek) tagol'odik. Ezeknek a m'elys'egeknek a sz'amoz'as'at, kezel'es'et a DOG automatikusan elv'egzi.

A lap form'atuma szab'alyozhat'o. Egy lap alap'ertelmez'es szerint a lapsz'ammal kezd'odik. A lapsz'amoz'as lehet folyamatos, fejezeten bel:ul folyamatos 'es el is maradhat. A sor szab'alyozhat'o hossz'us'ag'u, a bal marg'o 'allithat'o, meghat'arozhat'o a lapra ker:ul'o sorok sz'ama.

A lap azonos'it'asa k'etf'elek'eppen lehets'eges:

{{-C'el"ofejjel, ami azt jelenti, hogy a lap tetej'en ki'ir'odik egy azonos'it'o sz:oveg 'es a lapsz'am, 'es ez v'izszintes vonallal al'a lesz h'uzva. A lapsz'am mindig a lap k:uls'o sz'el'en jelenik meg, teh'at a bal oldali lapokon a sz:oveg el"ott, a jobboldaliakon pedig a sz:oveg ut'an;

{{-C'l'absorral, teh'at a lap alj'an v'izszintes vonal 'ir'odik ki ez al'a pedig a lap azonos'it'o sz:ovege.

A lap azonos'it'o sz:ovege p'aros lapon a dokument'aci'o c'ime, p'aratlan lapon a fejezet c'ime.

Ezek a szolg'altat'asok letilthat'ok.

\$cpt.

K:ul:onleges karakterek

A k:ul:onleges karaktereknek a k:ovetkez'o funkci'oik vannak:

{{-C{kifejez'eseket defini'alnak 'es azok feldolgoz'as'at adj'ak meg. Ezek a karakterek kifejez'esek soron bel:uli elhelyez'es'et is befoly'asolj'ak.

Ilyen a kifejez'es\tagol'o, a kifejez'es\:osszek:ot'o, a kifejez'es\elv'alaszt'o 'es a kifejez'es\deklar'ator;

((-(a kifejez'es poz'icion'al'asa. K'et olyan karakter van, amely befoly'asolja a kifejez'esek soron bel:uli kezd'o poz'ici'oj'at (a balra t:ort'en"o kiegyenl'it'est)). Ezek a karakterek: a tabul'ator 'es a bekezd'es kezdete;

((-(a teljes kifejez'es k:ul:onleges nyomtat'asa (a kifejez'es kihangs'ulyoz'asa).

\$sec.

Kifejez'estagol'o

A kifejez'esek tagol'as'ara minden esetben a sz'ok:oz karaktert haszn'aljuk. Tetsz'oleges sz'am'u sz'ok:ozt egyetlen sz'ok:oznek 'ertelmez:unk. Ez a karakter egy "helyettes'ithet"o sz'ok:ozt" reprezent'al, amit az output sz:ovegben legal'abb egy val'os'agos sz'ok:ozzel helyettes'it:unk att'ol f:ugg'o'en, hogy mennyi sz'ok:oz sz:uks'eges a sor jobbra igaz'it'as'ahoz.

\$pag/5.

(((\$ _Megjegyz'es:

((('Eppen ez'ert, ha g'epel:unk k:ovess:uk a szok'asos g'epel'esi st'ilust, a g'epel'esn'el teh'at a mondatok ut'an legal'abb egy sz'ok:ozt g'epelj:unk be 'es ugyan'igy g'epelj:unk sz'ok:ozt az :osszetettebb mondatokban a mondatr'eszeket hat'arol'o karakterek ut'an is.

\$sec.

Kifejez'es:osszek:ot"o

Ez a k:ul:onleges karakter lehet"ov'e teszi, hogy kett"o (vagy"t:obb) egym'as ut'an k:ovetkez"o sz'ot egyetlen " oszthatatlan " kifejez'esnek adjunk meg, azaz ezt a kifejez'est minden esetben egy sorban kell kinyomtatni. Alap'ertelmez'es szerint ez a legt:obb termin'alon a hull'amosvonal karakter, ASCII kodja a &7e, de ezt meg lehet v'altoztatni \$ctr paranccsal. Nyomtat'askor ez a karakter egy val'os sz'ok:ozzel helyettes'it"odik (ezen t'ulmen"oen az 'igy :osszek:ot:ott kifejez'esek nem szakadnak el egym'ast'ol, ha a sor jobbra kiigaz'it'odik).

(((\$ _Megjegyz'es:

((('Ezt a karaktert mind'ig olyan esetben haszn'aljuk, ha el szeretn'enk ker:sulni azt, hogy az output sor v'ege k'et (vagy

t:obb) sz'ot elv'alasszon egym'ast'ol, hiszen nem tudjuk, hogy az output sor t'enylegesen hol v'egz'odik.

\$pag/10.

\$sec.

Kifejez'eselv'alaszt'o

Ezzel a karakterrel megadhatjuk a kifejez'es lehets'eges elv'alaszt'as'anak a helyeit a sor v'eg'en. Standard esetben ez a backslash, ASCII k'odja &5c, de ezt meg lehet v'altoztatni a \$ctr paranccsal.

A karakter funkci'oja a k:ovetkez'o: ha a karakterrel l'etrehozott r'eszkifejez'es az aktu'alis soron m'eg elf'er, akkor a kifejez'es egybe fog kinyomtat'odni 'es a kifejez'eselv'alaszt'o hat'astalan lesz. A ki\fe\je\z'es ebben az esetben teljes el'g'e\sz'e\ben az aktu'alis sorba ker:ul. Ha azonban a kifejez'es\elv'alaszt'o m'ar nem helyezhet'o el az aktu'alis sorba, akkor csak az els'o r'eszkifejez'es fog az aktu'alis sorra ker:ulni az elv'alaszt'ojel (-) karakterrel egy:utt 'es a m'asodik r'eszkifejez'es a k:ovetkez'o sor elej'ere nyom\ta\l't'o\dik.

Pl.: ha egy sorban szerepel a sz'am'it'astechnika sz'o de csak 10 karakter hely van a sorban, nem f'er ki. Ha kifejez'es\elv'alaszt'oval adjuk meg, akkor az els'o r'esze (sz'am'it'as-) m'eg elf'er, s csak a t:obbi ker:ul a m'asik sorba, mint 9 sorral feljebb a "kife-jez'es" sz'on'al.

\$pag/5.

(\$_Megjegyz'es:

((A gyakorlati tapasztalatok alapj'an azt mondhatjuk, hogy a legt:obb nyelvben -- ahol a szavak 'erezheth'oan hossz'uak-- m'eg az is redund'ans lesz, ha a l'athat'oan hossz'u kifejez'esekre (szavakra) el"ore megadunk elv'alaszt'asokat. Enn'el sokkal jobb strat'egia az, ha a dokument'acio els'o gener'al'asa el"ott ezeket a szavakat m'eg nem v'alasztjuk el 'es a dokument'aci'o nyomtatott k'epe alapj'an ott adjuk meg ezeknek a szavaknak az elv'alaszt'as'at, ahol az eg'esz sz'o a k:ovetkez'o sorba ker:ult 'es 'igy az el"oz'o sorban nagy hely maradt :uresen. L'atni fogjuk, hogy ez nem t'ul gyakran fordul el"o, s"ot tudnunk kell, hogy egy j'ol

meg'irt dokument'aci'o mindig j'ol van szerkesztve is, 'es egy kifejez'es elv'alaszt'asa befoly'asolja azt is, hogy a j'ol kialakított sz:ovegszerkezet hogyan fog kin'ezni.

\$sec.

Kifejez'esdeklar'ator

Ezzel a deklar'atorral adhatunk meg olyan kifejez'eseket, amelyek tetsz"oleges karaktereket tartalmaznak, m'eg a k:ul:onleges karaktereket is (bele'ertve a sz'ok:ozt).

A kifejez'esdeklar'ator a "!" karakter

("a szaggatott f:ugg"oleges vonal", ASCII k'odja &7c).

Az els"o kifejez'es\deklar'ator adja meg a kezdet'et egy (fel nem oszthat'o) kifejez'esnek, a m'asodik pedig a v'eg'et. Ha (rendszerint g'epel'esi hiba miatt) a m'asodik (z'ar'o) karaktert elhagyjuk, akkor a sor v'ege lez'arja a kifejez'est. Ebb'ol k:ovetkezik, hogy egy sorn'al nagyobb kifejez'es nem adhat'o meg egyben!

Kifejez'es\deklar'atort 'altal'aban akkor haszn'alunk, amikor t:obb sz'ok:ozt tartalmaz'o kifejez'eseket 'irunk, ilyenkor a sz'ok:oz:ok sz'am'a g'epel'eskor r:ogz'ithet'o.

\$pag/10.

\$sec.

Tabul'ator

Ez a karakter lehet"ov'e teszi, hogy olyan poz'ici'okat (szimb'olikusan - nem sz'amszer:uen), adjunk meg,

amelyekre az input sz:ovegben hivatkozhatunk, azaz a tabul'ator\karakter

ut'ani sz:oveg a megadott poz'ici'ot'ol kezd"od"oen lesz elhelyezve. A "szimbolikus" sz'o azt jelenti, hogy a \$tab paranccsal az input file-ban 20 poz'ici'o be'all'it'as'ara van m'odunk 'es a tabu\l'a\tor\ka\rak\ter csak arra szolg'al, hogy ebb'ol az 'ert'ekhalmazb'ol a k:ovetkez"ot kiv'alasszuk. Alap'ertelmez'es szerint ez a karakter a bal kapcsos z'ar'ojel, ASCII k'odja &7b, de ezt is meg lehet v'altoztatni a \$ctr paranccsal. A tabu\l'a\tor\ka\rak\ter sohasem nyomtat'odik, kiv'eve a "k'ep" kezel'est. Ezt a karaktert mindig 'ugy 'ertelmezz:uk, mint kifejez'es\tagol'ot.

A tabul'ator a k:ul:onleges karakterekre a k:ovetkez"o hat'ast fejt ki:

{{-ca tabul'ator\karakter el"ott vagy ut'an l'ev"o

kifejez'es\összeköt'o 'es kifejez'es\elv'aszt'o karaktert nem veszi figyelembe (nem "fukcion'alnak" - mivel a funkci'ojuknak nincs jelent'ese)

{{-cha egy tabul'ator\karakter el"ott olyan kifejez'es van, amit egy kifejez'esdeklar'atorral adtunk meg, de a z'ar'o deklar'ator karakter hi'anyzik, akkor a tabul'ator z'arja le a deklar'aci'ot. A \$tab parancs a lehets'eges 'ert'ekeket sorban egym'as ut'an adja meg. Az 'ert'ekek elvileg n:ovekv"o sorrendben szerepelnek 'es minden tabu\\l'a\\tor\\ka\\rak\\ter ebb"ol az 'ert'eksorb"ol a k:ovetkez"o 'ert'eket v'alasztja ki, 'ugy ahogyan azt al'abb ismertetj:uk:

{{-ca bekezd'es v'ege ut'an (azaz ha a k:ovetkez"o input sor az els"o poz'ici'oban "\$" karaktert vagy tabul'ator\karaktert tartalmaz), a tabu\\l'a\\tor\\ka\\rak\\terrel t:ort'en"o poz'icion'al'as a \$tab paranccsal els"onek megadott 'ert'ekkel kezd"odik;

\$pag/5.

{{-cha a megadott 'ert'ekek k:oz:ul, m'ar mindet kiv'alasztottuk (azaz az aktu'alis bekezd'esben m'ar volt 20 db helyesen 'ertelmezett tabu\\l'a\\tor\\ka\\rak\\ter -- teh'at a tabul'ator aktu'alis poz'ici'oja a legutols'o lehets'eges 'ert'eken van), akkor hiba:uzenetet fogunk kapni 'es a tabul'ator aktu'alis poz'ici'oja v'altozatlan marad;

{{-cha a k:ovetkez"o 'ert'ek kisebb, mint a nyomtat'asi sor t'enyleges poz'ici'oja (azaz az el"oz"o kifejez'es olyan poz'ici'on v'egz"odik, ami a tabul'ator 'altal megjel'olt poz'ici'o ut'an van), akkor a tabul'ator aktu'alis poz'ici'oja azonos lesz a k:ovetkez"o bet"upozici'oval, azaz a tabul'ator sz'ok:ozk'ent fukcion'al.

{{{Felh'ivjuk a figyelmet arra, hogy a tabul'ator\karakter ut'an k:ovetkez"o sz'ok:oz:oket a sorkiegyenl'it'es figyelmen k'iv:ul hagyja.

\$pag/6.

{{-cha a tabul'ator ut'an a dokument'aci'o sz:ovege t:obb soron kereszt:ul folytatat'odik (azaz a bekezd'esben t:obb

sor van), akkor ez a sz:oveg a tabul'atorpoz'ici'o ut'an (balra igaz'itva) fog megjelenni. Azokat a bekezd'eseket 'irhatjuk le 'igy, amelyeknek nem kell a sor els"o poz'ici'oj'an kezd"odni (erre p'eldak'ent szolg'al a tabul'ator viselked'es'et le'ir'o im'enti felsorol'as. \$sec.

Bekezd'es v'ege, 'uj bekezd'es megad'asa

A sz:oveg t'ipus'u input sorokat

(amelyek nem tartalmaznak parancsot) egym'as ut'an l'ancoljuk, 'es ebb'ol a l'ancol'asb'ol 'all el"o egy bekezd'es eg'eszen addig, am'ig ezt le nem z'arja egy z'ar'o sor.

Egy ilyen (z'ar'o) sor vagy a "\$" (doll'ar) karaktert vagy tabul'atort tartalmaz az els"o poz'ici'oj'an, vagy pedig :ures. Ennek a "z'ar'o" sornak

a hat'asak'ent l'eterej:on az el"otte l'ev"o bekezd'es utols'o sora. Ez az utols'o sor nincs megny'ujtva, azaz nem lesz jobbra kiigaz'itva.

{Ime.

A "\$" vagy tabul'ator\karakterrel kezd"od"o sorok tov'abbi hat'asai:

{{-Ca tabul'ator\karakterrel kezd"od"o sorn'al a tabul'ator poz'ici'oja (aktu'alis poz'ici'o) a legut'obbi \$tab paranccsal megadott 'ert'ekek k:oz:ul az els"ore 'all'it'odik be (l'asd fentebb);

\$pag/3.

{{-Ca "\$" karakterrel kezd"od"o sor mindig lez'arja az el"oz"o bekezd'est a le'irtak szerint, 'es azut'an kezd"odik a ki'ert'ekel'ese (parancssor vagy :ures, stb.);

{{-Ca "\$"-al kezd"od"o sor 'es ennek a sornak a kezdete nem esik egybe egyetlen paranccsal sem, 'igy ez ism'et egy sz:oveg t'ipus'u sor (az 'uj bekezd'es els"o sora).

Ezen t'ulmen"oen ez a karakter befoly'asolja az els"o kifejez'es poz'ici'oj'at

ebben a bekezd'esben: ha ezut'an k:ozvetlen:ul egy kifejez'es k:ovetkezik

(pontosabban egy nem sz'ok:ozkarakter), akkor ez a kifejez'es a bal marg'o+5. poz'ici'on

fog kezd"odni.

Megjegyezzük, hogy ezzel a módszerrel az első kifejezést az 5. pozíciótól kezdve jobbra tolva helyezhetjük el, ha arra van szükségünk, hogy a bal margóhoz közelebb kerüljünk, akkor használjuk a tabulátort;

```
((-Ca "$" karakter a szövegtípusú sorban (az elsőo eset)
'ugy szerepel, mint az elsőo bekezdést lezáró karakter
('es meghatároz egy újat).
```

Ez csak akkor érvényes, ha az input sor elsőo pozícióján van a "\$".

A szövegtípusú sor belsejében ezt a karaktert is egy kifejezéshez tartozó karakternek vesszük (kivéve azt az esetet, amikor ez egy különleges kettős karakter elsőo karaktere);

```
((-Colyan input sor, amely csak a "$" karaktert tartalmazza
'es azt is
```

az elsőo pozíciójában (azaz a sor többi része üres), vagy a

sor teljesen üres.

Az ilyen sort úgy értelmezzük mint \$spa parancsot, azaz hatásként

egy üres sor jön létre a dokumentációban (lásd a \$spa parancsnál);

```
((-Ha a sor a "$" karakterrel kezdődik, a tabulátor
aktuális értéke 1 lesz (azaz a sorok bal oldala az elsőo
pozíciótól
```

kezdve lesz kiegyenlítve) és a tabulátorpozíciók

kiválasztása a tabulátor\karakterrel ismét a

legutóbbi \$tab parancs által megadott

értékek közül az elsőovel fog kezdődni;

```
((-Ca fent megadott szabály alól van kivétel (máskülönben
ez nem lenne szabály):
```

```
(((*Caz összes aláhúzást kijelölőo kettős karakter
```

```
(((*Ca $rg. indextel kijelölőo parancs
```

\$pag/10.

\$sec.

Különleges kettős karakterek kifejezések kiemelésére

Ezek a különleges kettős karakterek a kifejezés

k:ul:onleges nyomtat'as'at (kiemel'es'et) adj'ak meg.
Haszn'alatuk 'altal'anos felt'etelei a k:ovetkez"ok:

{{-{{a k:ul:onleges kett"os karakternek csak akkor van
k:ul:onleges 'ertelmez'ese, ha k:ozvetlen:ul egy kifejez'es
el"ott szerepel (ez kezdi a kifejez'est).

{{{*{{ha egy kifejez'est a kifejez'es\deklat'atorral adtunk
meg, akkor az els"o deklat'ator fogja jelezni az ilyen
kifejez'esnek a kezdet'et, azaz a
k:ul:onleges kett"os karaktert ez \$_el'e a deklar'ator el'e
kell g'epeln:unk;

{{{*{{ha k'et kifejez'es :ossze van k:otve a
kifejez'es\osszek:ot"ovel, akkor
ezek egyetlen kifejez'est alkotnak, így teh'at a k:ul:onleges
kett"os karaktert ezen k'et kifejez'es k:oz:ul az els"o el'e
kell helyezni:unk ('es ekkor a
teljes ":osszekapcsolt" kifejez'esre fog vonatkozni)

{{-{{ha a k:ul:onleges kett"os karaktert nem k:oveti kifejez'es
(csak :onmag'aban 'all'o kifejez'esk'ent szerepel),
akkor teljesen hat'astalan (nem vessz:uk figyelembe).

A k:ul:onleges kett"os karakterek 'alland'o 'ert'ekre vannak
el"ore defini'alva 'es ez nem v'altoztathat'o meg.

A kett"os karakterek 'es jelent'es:uk a k:ovetkez"o:

{{-{{doll'ar 'es al'ah'uz'as (\$ 'es _), k'odjai &24 'es &5f.
A kifejez'es

ugyanabban a sorban lesz al'ah'uzva mindaddig, m'ig egy
sz'ok:oz vagy sorv'ege karakter nem j:on.

Ez csak soron bel:uli
al'ah"uz'asra alkalmas.

Az al'ah'uz'as 'ugy t:ort'enik, hogy ugyanabban a sorban
(ahol az 'igy al'ah'uzott kifejez'es el"ofordul)
fel:ulnyomtatunk.

Az al'ah'uz'as a "_"
karakterrel t:ort'enik (al'ah'uz'as karakter, k'odja &5f);

{{-{{doll'ar 'es 2 al'ah'uz'as (\$, _ 'es _), k'odjai &24,
&5f 'es &5f.

A kifejez'es
al'a lesz h'uzva mindaddig, am'ig 'ujra a
doll'ar 'es al'ah'uz'as, al'ah'uz'as karaktersorozat nem
k:ovetkezik.

Ez lehet t:obbsoros al'ah'uz'as is.

Az al'ah'uz'as az el"obb le'irt m'odon t:ort'enik.

((-{doll'ar 'es kett"oskereszt (\$ 'es #), k'odjai &24 'es &23.

A kifejez'es

ugyanabban a sorban lesz k:ov'er'itve mindaddig, m'ig egy
sz'ok:oz vagy sorv'ege karakter nem j:on.

Szab'alyai azonosak az al'ah'uz'as szab'alyaival.

K:ov'er'it'es csak olyan sornyomtat'on lehets'eges,
melynek megfele"o parancsa.

((-{doll'ar 'es 2 kett"oskereszt (\$, # 'es #), k'odjai &24,
&23 'es &23.

A kifejez'es

k:ov'er'itve lesz mindaddig, am'ig 'ujra a
doll'ar 'es kett"oskereszt, kett"oskereszt karaktersorozat
nem k:ovetkezik.

Ez lehet t:obbsoros k:ov'er'it'es is.

((-{doll'ar 'es per-jel (\$ 'es /), k'odjai &24 'es &2f.

A kifejez'es

ugyanabban a sorban lesz d"olt be"us mindaddig, m'ig egy
sz'ok:oz vagy sorv'ege karakter nem j:on.

((-{doll'ar 'es 2 per-jel (\$, / 'es /), k'odjai &24, &2f
'es &2f.

A kifejez'es d"olt bet"us lesz mindaddig, am'ig 'ujra a
doll'ar 'es per-jel, per-jel karaktersorozat nem

k:ovetkezik.

Ez lehet t:obbsoros d"olt bet"us 'ir'as is.

\$sec.

'Ekezetes karakterek

Az 'ekezetes bet"uket 'ugy :utj:uk be, hogy a bet"u \$_el'e
(amelyre az

'ekezetest akarjuk tenni) be'irjuk a k:ovetkez"o karakterek
valamelyik'et:

((-{r:ovid 'ekezetek: ":" (kett"ospont) pl. :ul, :ol

((-Hosszu 'ekezetek: ' (apostrof) 'es " (macskak:or:om) pl.
'ir, 'ugy, 'ol, 'es, "or.
\$cpt.

A DOG parancsai

A DOG parancsai funkci'o szerint t:obb r'eszre oszthat'ok:

((-(Sz:ovegegys'egek defini'al'asa
(\$cpt, \$sec, \$prg, \$art, \$itm)

((-(Szerkesztett 'es k:oz'epre helyezett k'epparancsok
(\$pic, \$mid)

((-(T'abl'azatgener'al'o parancsok (\$stb, \$shd, \$vel, \$unl)

((-(K:ul:onleges feldolgoz'ast lez'ar'o parancs (\$eoc)

((-(Indext'etel azonos'it'asa (\$rg.)

((-(A sz:ovegegys'egek 'es lapform'atum le'ir'o parancsok
(\$par, \$ctr)

((-(Sz:oveg k:ozbeni vez'erl'o parancsok (\$pag, \$spa, \$tab)

((-(Egy'eb parancsok

(\$inp, \$tit, \$sum, \$reg, \$con, \$eli, \$nor, \$cnd, \$dou, \$nod,
\$rem, \$lhd)

\$cpt.

Sz:ovegegys'egek defini'al'asa

Mint m'ar az el"oz"oekben utaltunk r'a, sz:ovegegys'eknek a
fejezetszinteket nevezz:uk.

A sz:ovegegys'egek automatikusan m'elys'eg szerint
sorsz'amo:z'odnak.

\$mid/tc.

:Ot sz:ovegegys'eg (fejezet) szintet k:ul:onb:oztet:unk meg:

+---+-----+-----+-----+-----+

lssz!	n'ev	szint	sz'amosz'as
1.	cpt	fejezet	XX.
2.	sec	alfejezet	XX.YY.
3.	prg	bekezd'es	XX.YY.UU.
4.	art	cikkely	XX.YY.UU.WW.
5.	itm	t'etel	XX.YY.UU.WW.ZZ.

\$eoc.

\$spa/2.

\$_A^sz:ovegegy'segszintek^defini'al'o^parancsai:

{{ \$cpt / <sz'am> . [lapazonos'it'o sz:oveg]

{{ \$sec / <sz'am> .

{{ \$prg / <sz'am> .

{{ \$art / <sz'am> .

{{ \$itm / <sz'am> .

ahol a sz'am megadja, hogy a parancsot k:ovet"o h'any sor tartozik az adott sz:ovegegy'seg c'im'ehez.

A megadott sz'am'u

soron bel:ul semmif'ele parancsfigyel'es, sorsz'eth'uz'as nem

t:ort'enik, csak az 'ekezetes karakterek konvert'al'asa,

teh'at a

sorok v'altozatlan form'aban lesznek kinyomtatva.

A <sz'am> alap'ertelmez'ese egy.

A fejezetc'imek ki'ir'as'an'al a program minden esetben

lapot emel, ha a c'imet megel"oz"o :ures sorok, a c'imet

k:ovet"o :ures sorok, a c'imsor 'es 5 sor a fejezetb"ol nem fer m'ar re az aktu'alis lapra.

A fejezetszintet azonos'it'o parancs hat'as'ara automatikusan

l'trejon a megfelel"o fejezetsz'am.

A "\$cpt" parancsnak van egy k:ul:on param'etere, a

lapazonos'it'o

sz:oveg.

Minden lap alj'an vagy tetej'en k'erhet"o az

azonos'it'o sz:oveg ki'ir'asa.

Ha ezt nem adjuk meg a "\$cpt" parancs

param'eterek'ent, akkor a "\$cpt" c'im'et fogja a l'absorba vagy

az 'el"ofejbe

nyomtatni a p'aros lapsz'am'u oldalakon mindaddig, m'ig egy 'uj "\$cpt" parancs nem j:on.

(A l'absor ill. 'el"ofej nyomtat'asa a "\$par" parancsban letilthat'o.

Ebben a k:onyvben letiltottuk, ld. a 18. pont p'eld'aj'anak 3. sor'at.)

Minden sz:ovegegys'eg c'ime automatikusan r'esze a tartalomjegyz'eknek.

(Fel:ulb'ir'alhat'o a "\$par" parancsban.)

Egy sz:ovegegys'eg kezd"odhet:

((-Ca k:ovetkez"o p'aratlan lapon

((-Ca k:ovetkez"o p'aros lapon

((-Ca k:ovetkez"o lapon (ez az alap'ertelmez'es az els"o (cpt) szintre)

((-Caz aktu'alis lapon (ez az alap'ertelmez'es az als'o 4 szintre)

Ez szint'en fel:ulb'ir'alhat'o a "\$par" parancsban.

\$cpt.

K'ep megad'asa (\$pic)

Ezzel a paranccsal olyan egym'as ut'an k:ovetkez"o input, sz:oveg t'ipus'u

sorokat adhatunk meg, ahol egy input sornak a dokument'aci'o egy sora felel meg 'es egy input sorb'ol egy output sor keletkezik.

Ez t'enylegesen azt jelenti, hogy az input sorokat mindenfajta form'az'as n'elk:ul nyomtatjuk ki.

A parancs szintaxisa:

(\$pic[/<sz'am>,tcl.

A param'eterek jelent'ese:

((-Ca <sz'am> param'eter megadja a k'epet alkotj'o sorok sz'am'at.

Ez a sz'am ugyanakkor megadja a dokument'aci'o nyomtatott form'aj'aban a k'ep sorainak a sz'am'at is.

Ennek

alap'ertelmez'es szerinti 'ert'eke &7FFF, 'igy teh'at, ha nem adjuk meg a sz'amot, akkor a k'ep addig fog tartani, am'ig az inputon egy "z'ar'o"

parancs nem lesz (ezt l'asd k'es"obb);

((-(a tc menmonik megad'asa azt

jelzi, hogy az 'ekezetes karaktereket 'at kell k'odolni.

A tc param'eter megad'asa n'elk:ul a kifejez'esek (input sorok)

abban a form'aban fognak

kinyomtat'odni, ahogy az inputon szerepelnek.

A k'ep l'etrehoz'as'anak a szab'alyai a k:ovetkez"ok:

((-(a k'epet alkot'o input sorok nem lehetnek parancssorok, ez al'ol

kiv'etel a \$inp parancs, amelynek v'egrehajt'as'at a k'epen bel:ul

elfogadjuk (a r'eszleteket l'asd az input file-ok vez'erl'es'en'el).

Ez a

tulajdons'ag lehet"ov'e teszi, hogy forr'asprogramot

m'asoljunk a

dokumentum sz:oveg'ebe.

\$pag/6.

((((P'elda (a z'ar'ojelek k:oz:ott megjegy'ez'esek vannak):

(((((\$pic.

(k'ep megad'asa)

(((((\$inp/source.

(a forr'asprogramot tartalmaz'o file)

(((((\$eoc.

(a k'ep lez'ar'asa)

\$pag/8.

((-(a k'ep nyomtat'asa az al'abbi k'et esetben fog befejez"odni:

(((* (a megadott <sz'am>- 'u sz:ovegt'ipus'u input sor feldolgoz'asa

befejez"od:ott;

{{(*{az input sorok k:oz:ott parancs t'ipus'u sor van
(kiv'eve a \$inp parancsot).

Ennek a m'odszernek a haszn'alatakor k:otelez"o, hogy a z'ar'o
parancs a \$eoc parancs legyen (ez a parancs mindig lez'ar
n'eh'any

k:ul:onleges feldolgoz'ast.

Ezt k'engelmes

haszn'alni, mivel a k'epet alkot'o sorokat nem kell

lesz'amolnunk --

elegend"o, ha el'eg nagyra adjuk meg a <sz'am>-ot, vagy

meghagyjuk annak

az implicit 'ert'ek'et, 'es a k'epmegad'ast egy m'asik
paranccsal lez'arjuk);

A k'epnek egy speci'alis v'alfaja a k:oz'epre helyezett k'ep.
Megad'asa a \$mid paranccsal lehets'eges.

A

k'epet alkot'o sorokat a DOG 'ugy helyezi el, hogy
szimmetriatengely:uk

a sor k:ozepe legyen.

Ezt f"oleg a c'imlapon l'ev"o elnevez'esek megad'asakor
c'elszer"u haszn'alnunk.

A parancs szintaxisa:

{ \$mid[/][inl=]<sz'am>,tc}.

A k:oz'epre helyezett k'ep l'etrehoz'as'anak param'eterei 'es
szab'alyai

pontosan

megegyeznek azzal, amit az el"oz"oleg a k'epn'el elmondtunk.

Csak a

"k'ep" hely'ere kell minden:utt "k:oz'epre helyezett k'ep"-et
gondolnunk, a

"\$pic" helyett pedig "\$mid"-et.

\$cpt.

A k:ul:onleges sz:ovegfeldolgoz'asok megad'as'anak a v'ege

Ez a parancs rendszerint azt jel:oli, hogy a

sz:ovegt'ipus'u input sorok

k:ul:onleges feldolgoz'as'at be kell fejezni, azaz ez a parancs

z'ar le n'eh'any k:ul:onleges feldolgoz'at megad'o parancsot.

Ezzel a paranccsal lehet lez'arni a \$pic (k'ep), a \$mid

(k:oz'epre helyezett k'ep), a \$shd (t'abl'azat fejl'ece),
a \$stb (t'abl'azat) parancsokat - l'asd m'eg a
megfelel"o parancsok le'ir'as'at is.

Ha ezt a parancsot nem a fenti sz:ovegk:ornyezetek
valamelyik'eben
haszn'aljuk, akkor nincs hat'asa.

A parancs szintaxisa:

{ \$eoc.

\$cpt.

T'abl'azatok gener'al'asa

A DOG egyik igen k'enyelmes szolg'altat'asa a t'abl'azatok
l'etrehoz'as'an'al a t'abl'azat gener'al'asa.

T'agabb 'ertelemenben t'abl'azatnak nevezz:uk azt a ter:uletet,
amely be

van keretezve 'es f:ugg"oleges vonalakkal oszlopokra van
osztva, ezen oszlopok k:oz:ott pedig valamilyen sz:oveg van.

Ezt a keretez'est,

oszlopokra tagol'ast, az oszlopon bel:uli sz:ovegelhelyez'est,
fejl'eckezel'est lehet a DOG t'abl'azatgener'al'o parancsaival
automatiz'alni.

Egy t'abl'azat megad'as'anak 'altal'anos szerkezete az input
file-ban:

{ \$stb.((((t'abl'azat megad'asa)

{ \$shd.((((t'abl'azat fejl'ec'enek megad'asa)

{ \$vel/.....((((a fejl'ec f:ugg"oleges vonalainak poz'ici'oja)

{ (((a fejl'ec sz:ovege)

{ \$eoc.((((a fejl'ec v'ege)

{ \$vel/.....((((a t'abl'azat f:ugg"oleges vonalainak

poz'ici'oja)

{ (((a t'abl'azat sz:ovege)

{ \$eoc.((((a t'abl'azat v'ege)

Vizsg'aljuk meg az egyes parancsok pontos jelent'es'et.

\$sec.

A t'abl'azat kezd"o parancsa

A parancs form'aja:

{ \$stb[/<[num=<[sorok>],[vch=<f:ugg"oleges-vonal>],?

((([hch=]<vizszintes-vonal>,[cch=]<metsz'espontok>),?
((([tc,unl,spa,nsbv,nsph].

\$pag/5.

A param'eterek jelent'ese:

((([num(((megadja, hogy h'any sor ut'an kell a t'abl'azatban egy
v'izszintes
elv'alaszt'ovonalat (automatikusan) beilleszteni.
Ez lehet"os'eget ad arra,
hogy a t'abl'azatot k'et r'eszre osszuk fel.
Alap'ertelmez'es szerinti
'ert'eke &7FFF, ami gyakorlatilag azt jelenti, hogy nem ker:ul
sor ilyen
beilleszt'esre;

\$pag/3.

((([spa(((az elv'alaszt'o sor egy :ures sor legyen, azaz olyan
sor, amelyikben nincsenek t'etelek, csak f:ugg"oleges vonalak.
Alap'ertelmez'es szerint ez 'erv'enyos;

((([unl(((az elv'alaszt'o sor egy v'izszintes vonal legyen.
A spa 'es unl parancsok k:olcs:os:osen kiz'arj'ak 'egym'ast,
egyszerre csak az egyik szerepelhet.
Ha m'egis mindkett'o meg van adva, az
'erv'enyos parancs az lesz, amelyik a parancssorban k'es"obb
van;

((([vch(((ebb"ol a karakterb"ol lesz a f:ugg"oleges oszlop.
Alap'ertelmez'es
szerint erre a "!"-et (a f:ugg"oleges szaggatott vonal)
haszn'aljuk;

((([hch(((ebb"ol a karakterb"ol lesz a v'izszintes vonal.
Alap'ertelmez'es szerint ez a "-" (elv'alaszt'ojel) karakter;

((([cch(((a metsz'espontokban haszn'alt karakter, amikor
a f:ugg"oleges 'es v'izszintes vonalak metszik egym'ast.
Alap'ertelmez'es szerint ez a "+" (:osszead'asjel);

((([nsbv(((a t'abl'azat f:ugg"oleges vonalaihoz 'all'itja a
sz:oveget

a t'abl'azaton bel:uli elhelyez'esi m'od f:uggv'eny'eben,
amelyet a \$vel parancs hat'aroz meg (ld. 8.3 pont).
Alap'ertelmez'es szerint egy sz'ok:oz kimarad;

{-nsph{{k:ozvetlen:ul a v'izszintes vonalakhoz 'all'itja a
sz:oveget mivel az alap'ertelmez'es szerint egy-egy sort
kihagy;

{-tc{{a t'etelek (kifejez'esek) 'atk'odol'asra ker:ulnek.
Alap'ertelmez'es szerint nem t:ort'enik 'atk'odol'as.

\$pag/10.

\$sec.

A t'abl'azat fejl'ec'enek a megad'asa (\$shd)

Ez a parancs lehet"ov'e teszi a t'abl'azat fejl'ec'enek,
azaz a t'abl'azat azon r'esz'enek megad'as'at, amellyel a
t'abl'azat oszlopait azonos'itjuk.

Ennek a fejl'ecnek meg kell jelennie minden olyan
lapon, ahol a t'abl'azat folytatat'odik (ha a t'abl'azat
hosszabb egy oldaln'al).

\$pag/3.

A parancs szintaxisa:

{shd.

A fejl'ec l'etrehoz'as'anak a szab'alyai:

{{-Ca fejl'ec elhagyhat'o;

{{-Cegy t'abl'azatban tetsz"oleges sz'amu fejl'ec, tetsz"oleges
helyen megadhat'o;

{{-Ca fejl'ec megad'as'at a \$eoc paranccsal vagy egy
m'asik \$shd paranccsal z'arhatjuk le;

{{-Cha a t'abl'azat t:obb lap hossz'us'ag'u, akkor a
legutolj'ara megadott fejl'ec fog automatikusan ism'etl"odni
minden lap tetej'en.

Ez a k:ovetkez"o korl'atoz'asokat eredm'engezi:

{{*Ca fejl'ecen bel:ul nincs lapv'alt'as

{{{*{a \$inp parancs haszn'alata a fejl'eben nem megengedett,

{{{*{ha a fejl'ec f:ugg"oleges vonalainak poz'ici'oi elt'er"oek
a t'abl'azat t:obbi r'esz'eben l'ev"o f:ugg"oleges
vonalak poz'ici'oit'ol, akkor \$_k:otelez"o
haszn'alnunk a \$vel parancsot.

{{-{ha net'an a t'abl'azaton bel:ul ujjabb fejl'ecet nyitunk,
akkor automatikusan egy v'izszintes z'ar'ovonal nyomtat'odik ki.

\$pag/2.

{{-{a fejl'ec defin'ici'oj'anak a v'ege (\$eoc) ujjabb v'izszintes
vonalat 'ir ki;

\$pag/3.

{{-{azokban az oszlopokban, ahol nem adtunk meg semmilyen
kiegyenl'it'est a \$vel parancssal, ott a fejl'ec minden t'etele
"k:oz'epre" ker:ul (l'asd a \$vel parancsot);

\$pag/5.

\$sec.

A t'abl'azat jellemz"oinek megad'asa (\$vel)
Ezzel a parancssal megadhatjuk a f:ugg"oleges vonalak
poz'ici'oit 'es a t'abl'azatt'etelek elhelyez'es'et.
Minden t'abl'azatban el"osz:or ennek a parancsnek kell
szerepelnie, 'es csak ezt k:ovetheti a t'etelek megad'asa.

\$pag/12.

A parancs form'atuma:

(\$vel[/({([p1=]<p1>,[p2=]<p2>,[p3=]<p3>,[p4=]<p4>,[p5=]<p5>,
({({([p6=]<p6>,[p7=]<p7>,[p8=]<p8>,<p9=]<p9>,
({({([p10=]<p10>,[p11=]<p11>,[p12=]<p12>,[p13=]<p13>,
({({([p14=]<p14>,[p15=]<p15>,[p16=]<p16>,
{[/[just;][j1=]<j1>,[j2=]<j2>,[j3=]<j3>,[j4=]<j4>,[j5=]<j5>,
({({([j6=]<j6>,[j7=]<j7>,[j8=]<j8>,[j9=]<j9>,
({({([j10=]<j10>,[j11=]<j11>,[j12=]<j12>,[j13=]<j13>,
({({([j14=]<j14>,[j15=]<j15>,allr allm alll].

A param'eterek jelent'ese:

{{-{a P1-P16 param'eterek megadj'ak a f:ugg"oleges vonalak

poz'ici'oit (a poz'ici'ok 1-t"ol kezd"odnek).
Maxim'alisan 16 f:ugg"oleges vonalat lehet
megadni (ez pedig 15 oszlopot hat'aroz meg).

(((*A poz'ici'ok 'ert'ekeinek eleget kell tenni:uk
a k:ovetkez"o korl'atoz'asoknak:

(((*{az 'ert'ekeknek n:ovekv"o sorrendben kell szerepelni 'ugy,
hogy a
k'et egym'as melletti 'ert'ek 'altal megadott poz'ici'ok
k:oz:ott elhelyezked"o oszlop m'ereite akkora legyen, hogy
legal'abb egy nulla
hossz'us'ag'u t'etel elf'erjen benne (ami azt jelenti, hogy a
k:ovetkez"o poz'ici'o 'ert'eke, ha nem szerepel a t'abl'azat
param'eterei
k:oz:ott az nspv param'eter (\$stb parancs), akkor legal'abb
h'arommal nagyobb
'es ha az nspv-t haszn'altuk, akkor legal'abb egygyel nagyobb).
Az a t'etel, ami egy ilyen ":ures" oszlophoz
tartozik nyilv'anval'oan egy :ures t'etel kell hogy, legyen
(l'asd k'es"obb);

(((*{a megadott poz'ici'ok (v'izszintesen) nem lehetnek
nagyobbak, mint a sor m'ereite;

(((*{legal'abb k'et poz'ici'ot meg kell adnunk;

((-{a JUST alszekci'o J1-J15 param'eterei adj'ak meg
oszloponk'ent a t'etelek elhelyez'es'enek m'odj'at.
Itt az al'abbi 'ert'ekeket adhatjuk meg:

(((*{Jx=L (balra) - a t'etel "balra-igaz'it'odik", azaz az
oszlop bal oldali f:ugg"oleges vonal'ahoz (a vonal mellett
egy sz'ok:oz kimarad, ha az nspv param'eter nem szerepel
a \$stb parancsban).

Ez az 'ert'ek alap'ertelmez'es
a t'abl'azaton bel:ul, azaz a t'abl'azat fejl'ec'en k'iv:ul;

(((*{Jx=M (k:oz'epre) - a t'etel az oszlopok k:oz:ott k:oz'epen
lesz elhelyezve.

A t'abl'azat fejl'ec'eben ez az alap'ertelmez'es;

(((*Jx=R (jobbra) - a t'etel "jobbra igaz'it'odik", azaz az oszlop jobb oldali f:ugg"oleges vonal'ahoz (most is, mint a balra igaz'it'asn'al, egy sz'ok:oz kimarad);

(((*Az ALLL, ALLR, ALLM param'eterek egyszerre adj'ak meg a kiigaz'it'ast az :osszes oszlopra.

Alkalmaz'asuk nem z'arja ki azt, hogy a J1-J15 param'etereket is megadjuk, azaz 'ertelmez'es:uk a k:ovetkez"o: "az :osszes oszlop" L (balra - ALLL), R (jobbra - ALLR) vagy M (k:oz'epre - ALLM) lesz kiigaz'itva, kiv'eve azokat az oszlopokat, amelyekre a kiigaz'it'ast k:ul:on megadtuk a J1-J15 param'eterekkel.

Ha egy t'abl'azat nem tartalmaz a fejl'ecen kiv:ul \$vel parancsot, akkor a f:ugg"oleges vonalak poz'ici'oja ugyanaz lesz a t'abl'azat belsej'eben, mint a t'abl'azat fejl'ec'eben.

Mivel a kiigaz'it'asokat nem adtuk meg ebben az esetben (a fejl'ecben) a J1-J15 param'eterekkel sem pedig az ALLL, ALLLR, ALLLM param'eterekkel, 'igy most ezek 'ert'eke L (balra) lesz ('es a fejl'ecben, ha nem adjuk meg a kiigaz'it'ast akkor annak 'ertelmez'ese M (k:oz'epre).

\$pag/12.

Hogy jobban meg'erts:uk, l'assunk erre k'et p'eld'at:

1. P'elda:

(((\$stb.

(((\$shd.

(((\$vel/1,8,23/,r.

(((<a fejl'ec t'etelei>

((<eoc.

(((<a t'abl'azat t'etelei>

((<eoc.

((Az 1, 8, 23.

poz'ici'okn'al h'uz f:ugg"olegest.

Az els"o

oszlopban a kiigaz'it'as M (alap'ertelmez'es szerinti), a m'asodikban R (a parancs 'altal megadott helyen) lesz,

a t'abl'azatban, vagyis a fejl'ecen k'iv:ul pedig az
alap'ertelmez'es szerinti
L 'ert'ek hat'arozza meg a sz:oveg 'all'it'as'at.

\$pag/14.

2. P'elda:

(((\$stb.

(((\$shd.

(((\$vel/1,8,15,18,23.

(((\$a fejl'ec t'etelei>

(((\$eoc.

(((\$shd.

(((\$vel/1,8,23/alll.

(((\$a fejl'ec t'etelei>

(((\$eoc.

(((\$vel/1,23/allm.

(((\$a t'abl'azat t'etelei>

(((\$eoc.

\$pag/10.

(((\$A p'elda egy k'etfejl'eces megold'ast mutat be, l'assuk egy
kis rajzon hogy is n'ez ez ki.

\$pic/tc.

```

+-----+-----+-----+
|         |         |         |         |
+-----+-----+-----+
|         |         |         |         |
+-----+-----+-----+
|         |         |         |         |
+-----+-----+-----+
|         |         |         |         |
+-----+-----+-----+

```

<== els"o fejl'ec

<== m'asodik fejl'ec

<== a t'abl'azat teste

\$eoc.

\$pag/4.

(((\$Az els"o fejl'ecben minden t'etel az alap'ertelmez'es szerint
k:oz'epre van igaz'itva, a m'asodikban az :osszes t'etel balra
igaz'it'as'at 'irja el"o az "alll" param'eter, m'ig a t'abl'azat
test'eben minden k:oz'epre lesz igaz'itva.

\$sec.

A t'abl'azat t'eteleinek megad'asa

(((\$a t'eteleket tabu\l'a\tor\ka\rak\terek k:oz'e tessz:uk;


```
{{-{{elvilleg azok a sz:ovegt'ipus'u input sorok, amelyek egy
t'abl'azaton bel:uli sort adnak
meg (legt:obbsz:or ez egy input sort jelent), annyi
tabul'ator\karaktert tartalmaznak, mint amennyi f:ugg"oleges
vonalat megadtunk a legut'obbi $vel
parancs seg'its'eg'evel (a p'eld'akat l'asd k'es"obb);
```

```
{{-{{egy t'abl'azatbeli t'etelt alkot'o kifejez'es k'et egym'as
ut'an k:ovetkez"o tabul'atorjel k:oz:ott kell legyen, ebben a
kifejez'esben tetsz"oleges karakterek szerepelhetnek ('igy
sz'ok:oz is lehet, ami lehet"ov'e teszi azt, hogy egy t'etelt
a t'abl'azat rubrik'aj'aban a
megadott kiigaz'it'ast'ol elt'er"oen helyezze:unk el);
```

```
{{-{{sz:uks'eg:unk lehet arra is hogy adott oszlopokban ":ures"
t'eteleket adjunk meg, ebben az esetben ez a t'etel
egyszer:uen kimarad a sz:ovegt'ipus'u input sorb'ol (azaz a
sorban ezen
a helyen egym'as mellett lesz k'et tabul'ator\karakter).
```

```
{{(P'elda (ahol a "<" karakter az 'erv'enyos
tabul'ator\karakter):
```

```
{{{{($stb.
{{{{($vel/1,10,20,30.
{{{{{{(els"o<<harmadik<
{{{{{{(els"o<m'asodik<
{{{{($eoc.
```

```
$pag/4.
```

```
{{(Az els"o
```

```
esetben nem adtunk meg t'etelt a m'asodik oszlopba, a m'asodik
esetben pedig a t'abl'azat harmadik oszlop'ara nem volt t'etel
megadva.
```

```
Az oszlopoknak megfelel'o t'etelek :uresek maradnak;
```

```
$pag/3.
```

```
{{-{{A program nem t:ordeli t:obb sorba a hossz'u
t'eteleket, azaz egy t'etel hossza legfeljebb akkora lehet, mint
a sz'am'ara lefoglalt oszlop!
```


\$pag/5.

{{{\$_Megjegyz'es:

{{{{{Ha egy t'etel l'athat'oan t:obb sort ig'enyel a
t'abl'azatban

(pl. a fejl'ecben), akkor a felhaszn'al'ora h'arul,
hogy t:obb input sorban biztos'itson helyet a t'etelnek.

{{{-{nem adhatunk meg t:obb t'etelt, mint amennyi oszlop van a
t'abl'azatban.

Ha ezt figyelmen k'iv:ul hagyjuk, akkor hiba:uzenetet kapunk
'es a

"t:obblet" t'etel megad'as'at a program nem veszi figyelembe.

\$pag/5.

\$sec.

Vizszintes vonal h'uz'asa

A \$unl parancs egy v'izszintes sort hoz l'etre a
t'abl'azatban ott, ahol a parancsot kiadtuk.

A parancs form'atuma:

[\$unl[/<sz'am> spa].

A param'eterek jelent'ese:

{{{*{a <sz'am> param'eter hat'as'ara l'etrej:on a k'ert sz'am'u
v'izszintes sor a

t'abl'azatban 'es ezen sorok el"ott 'es ut'an egy :ures sor is
nyomtat'odik (att'ol f:ugg"oen, hogy a \$shd parancs nsph
param'eter'et haszn'altuk-e vagy sem).

Ha nem adunk meg param'etert, akkor egy v'izszintes sort h'uz;

\$pag/5.

{{{*{az spa param'eter a v'izszintes vonal f:ol'e 'es al'a egy
:ures sort irat ki.

Sohasem haszn'alhat'o a <sz'am>

param'eterrel egy:utt.

\$pag/10.

1. P'elda:

{{{\$stb/nsph.

{{{\$shd.

{{{\$vel/1,30,60.


```

(((a fejl'ec t'eteleinek megad'asa>
(($eoc.
(($vel/1,30,45,60.
(($unl.
(((a t'abl'azat t'eteleinek a megad'asa>
(($eoc.

```

'igy a fejl'ec ut'an k'et v'izszintes vonal lesz.
\$pag/5.
\$sec.

A t:obbi parancs jelent'ese a t'abl'azaton bel:ul
Elvileg minden parancsot haszn'alhatunk a t'abl'azatdefin'ici'on
bel:ul,
az input file-ban haszn'alhat'o parancsok k:oz:ul.
Term'eszetesen n'eh'any
haszn'alata bizonyos k:ornyezetben tiltott 'es n'eh'any parancs
'ertelmez'ese egy kiss'e elt'er att'ol, mint ahogyan a
szabv'anyos sz:oveg gener'al'askor v'egrehajt'odik.
Most ismertetj:uk a kiv'eteleket 'es azokanak a parancsoknak
a r'eszletes szintaxis'at, amelyekn'el a t'abl'azat megad'asa
szempontj'ab'ol erre sz:uks'eg van (ha a szintaxist nem
ismertetj:uk, akkor az megtal'alhat'o
az adott parancs le'ir'as'an'al):

```

((-Ca sz:ovegegys'egek megad'as'at szolg'al'o parancsok
($cpt, $sec, $prg, $art, $itm) 'es a k'epdefini'al'o parancsok
($pic,$mid)
haszn'alata $_tilos a t'abl'azaton bel:ul;

```

((-Ca \$eoc parancs megad'asi k:ornyezet f:uggv'eny'eben lez'arja
a fejl'ec vagy az eg'esz t'abl'azat megad'as'at.
A fejl'ec megad'as'anak lez'ar'as'ara
'es a t'abl'azat lez'ar'as'ara mindig ezt a parancsot kell
haszn'alnunk;

\$pag/4.

((-Ca \$inp parancsot (a "be'agyazott" input file megad'as'ara
szolg'al -- l'asd az input file-ok vez'erl'es'en'el) a fejl'ec
megad'asakor tilos haszn'alni, a t'abl'azaton bel:ul azonban
haszn'alata megengedett, 'es v'egre is hajt'odik;

((-Ca \$pag parancs (felt'eteles vagy felt'etel n'elk:uli

lapv'alt'as) jelent'ese a szok'asos, de a hat'asak'ent
l'etrej:ov"o lapv'alt'as megt:ori a t'abl'azatot, azaz
lez'arja a t'abl'azatot az aktu'alis lapon
(ha a t'abl'azatnak legal'abb egy sora kinyomtat'odott).
A t'abl'azat lez'ar'asa egy v'izszintes vonal ki'irat'as'at 'es
a legut'obb megadott fejl'ec 'uj lapra kiirat'as'at jelenti.
A DOG a lapdob'ast is v'egrehajtja, ha az aktu'alis lap betelt;

\$pag/5.

{{(-{az :ures sorok megad'as'at szolg'al'o parancs (\$spa)
'ertelmez'ese nem

v'altozik meg, csak annyi a k:ul:onbs'eg, hogy megjelennek a
t'abl'azat f:ugg'oleges vonalai.

\$cpt.

A lap 'es sz:ovegegys'egek form'atum'at le'ir'o parancsok

A lap m'eret'et 'es form'atum'at, valamint a sz:ovegegys'egek
form'atum'at a "\$ctr" 'es "\$par" parancsok k:ul:onb:oz'o
param'etereivel adhatjuk meg.

\$sec.

Lapm'eret defini'al'asa

A lap m'eret'et a "\$ctr" parancssal lehet kijel:olni, illetve a
sz:ovegben megv'altoztatni.

A parancs form'aja:

\$pic/tc.

{{(\$ctr/[lnb=]<lapm'eret>,[cnb=]<sorm'eret>,
[mar=]<bal marg'o poz.>,
tab=<tabu\\l'a\\tor\\ka\\rak\\ter>,
exc=<kifejezes osszekoto karakter>.

\$eoc.

A param'eterek jelent'ese:

{{(-{lnb=<lapm'eret>

{{{{(A lapon elhelyezend'o sorok sz'am'at hat'arozza meg.

Ebbe

belesz'am'it a lapsz'amoz'as sora, az ut'ana el"o'ir'as szerint
kihagyand'o sorok sz'ama, a lapalji al'ah'uz'as 'es az el"otte
kihagyand'o sorok sz'ama valamint az als'o c'imsor is.

Ezek levon'asa ut'an kapjuk meg a t'enylegesen ki'ir'od'o
sz:ovegsorok sz'am'at.

\$pag/10.

{{{-{cnb=<sorm'eret>

{{{{Ezzel a param'eterrel hat'arozhatjuk meg az egy sorba ker:ul"o karakterek sz'am'at, vagyis a lap sz'eless'eg'et.

A be'all'itott

bal marg'o pozici'o 'ert'eke ebb"ol levon'odik,teh'at a t'enyleges karaktersz'amot a sorm'eret-(bal~marg'o~pozici'o) adja meg.

Ha pl. a sorm'eret 70, a bal marg'o pozici'oja 5, akkor a sorba maximum 65 karakter ker:ulhet.

\$pag/5.

{{{-{mar=<bal marg'o pozici'o>

{{{{A lap bal sz'els"o karakterpozici'oj'anak be'all'it'asa a feladata.

A lap sorainak els"o karaktere ebbe a pozici'oba ker:ul.

\$pag/5.

{{{-{tab=<tabu\l'a\tor\ka\rak\ter>

{{{{Az a karakter, amely tabu\l'a\tor\ka\rak\terk'ent fog funkcion'alni a sz:ovegben.

Alap'ertelmez'ese a nyit'o kapcsos z'ar'ojel.

\$pag/5.

{{{-{exc=<kifejez'es:osszek:ot"o karakter>

{{{{Az a karakter, amely kifejez'es:osszek:ot"o karakterk'ent fog funkcion'alni a sz:ovegben.

Alap'ertelmez'ese a hull'amos vonal.

\$pag/5.

\$sec.

A lap 'es sz:ovegegys'eg form'atum'anak be'all'it'asa

A lapok 'es a sz:ovegegys'egek c'imeinek form'atuma adhat'o meg a "\$par" paranccsal.

Ha sz:oveg k:ozben fordul el"o,

megv'altoztatja az el"oz"oleg defini'alt form'atumot.

\$pag/10.

A parancs form'aja:

```
((($par/([pos=]<a sz:ovegegys'eg c'im'enek elhelyez'ese>,  
([sbc=]<:ures sorok sz'ama a sz:ovegegys'eg c'ime el"ott>,  
([sac=]<:ures sorok sz'ama a sz:ovegegys'eg c'ime ut'an>,  
([for=]<a c'im 'ir'asi m'odja>,  
([sbp=]<a lapsz'amot k:ovet"o :ures sorok sz'ama>,  
([sap=]<a l'absort megel"oz"o :ures sorok sz'ama>,  
([nou],[noi],[cpl],[rci],[spn],[nps],  
([pag=<sz'am>],  
([cpt=<sz'am>]).,
```

\$pag/8.

A param'eterek jelent'ese:

```
((-(pos = a sz:ovegegys'eg c'im'enek elhelyez'ese  
((A k:ovetkez"o sz:ovegegys'eg kezd'esi hely'et hat'arozza  
meg.  
((A lehets'eges 'ert'ekek:
```

```
(((*op: (A sz:ovegegys'eg a k:ovetkez"o p'aratlan sz'am'u  
lap tetej'en kezd"odik (odd page)  
(((*ep: (A sz:ovegegys'eg a k:ovetkez"o p'aros sz'am'u  
lap tetej'en kezd"odik (even page)  
(((*np: (A sz:ovegegys'eg a k:ovetkez"o lap tetej'en kezd"odik  
(next page)  
(((*no: (A sz:ovegegys'eg nem ker:ul 'uj lapra (no page).
```

```
((A pos param'eter egy listat'ipus'u param'eter.
```

Ez azt

jelenti, hogy az egym'as ut'an felsorolt (max. 5) 'es egym'ast'ol "-" jellel el'vasztott 'ert'ekek rendre az adott m'elys'eg"u sz:ovegegys'egre vonatkoznak.

Pl.: a \$par/op-op-np-ep-no.

parancs hat'as'ara az els"o k'et fejezetm'elys'eg (sz:oveg\egys'eg szint)

mindig a k:ovetkez"o p'aratlan lapon, a harmadik szint az aktu'alisan k:ovetkez"o lapon, a negyedik a k:ovetkez"o p'aros sz'am'u lapon, az :ot:odik pedig az aktu'alis lapon fog kezd"odni.

((A lap'ertelmez'es szerint az els"o szint a k:ovetkez"o lapon, a t:obbi pedig az aktu'alis lapon kezd"odik.

{{-({sbc = :ures sorok sz'ama a sz:ovegegys'eg c'ime el"ott
{{{Minden sz:ovegegys'eg c'ime el"ott ennyi :ures sor fog
nyomtat'odni.

Szint'en lista\t'ipus'u param'eter, teh'at minden sz:ovegegys'eg
m'elys'egre k:ul:on megadhat'o.

{{{{Alap'ertelmez'ese: minden szinten a c'im el"ott k'et :ures
sor
marad ki.

{{-({sac = :ures sorok sz'ama a sz:ovegegys'eg c'ime ut'an
{{{Minden sz:ovegegys'eg c'ime ut'an ennyi :ures sor fog
nyomtat'odni.

Ez is lista\t'ipus'u param'eter.

{{{{Alap'ertelmez'ese: minden szinten a c'im ut'an k'et :ures
sor marad ki.

\$pag/7.

{{-({for = a c'im 'ir'asi m'odja

{{{{A sz:ovegegys'eg c'im'enek ki'ir'asi m'odj'at hat'arozza
meg.

Ez is lista\t'ipus'u param'eter.

{{{{Lehets'eges 'ert'ekei:

{{{{{(-{no(a c'im al'ah'uz'as n'elk:ul lesz ki'irva

{{{{{(-{un(a c'im al'ah'uzva lesz ki'irva

{{{{Alap'ertelmez'ese: minden c'im al'a lesz h'uzva

{{-({sbp = a lapsz'amot k:ovet"o :ures sorok sz'ama

{{{{Alap'ertelmez'ese: k'et :ures sor k:oveti a lapsz'amos'ast.

{{-({sap = a l'absort megel"oz"o :ures sorok sz'ama

{{{{Alap'ertelmez'ese: k'et :ures sor el"ozi meg a lapalji
vizszintes sort.

{{-({nou

{{{{A lapok alj'ara nem kell v'izszintes vonalat h'uzni.

Ez alap'ertelmez'esben nyomtat'odik.

{{-({noi

{{{{Nem kell l'absort nyomtatni.

Alap'ertelmez'esben minden p'aros

lap alj'ara a "\$tit" parancsban megadott f"oc'im, minden

p'aratlan lap alj'ara az aktu'alis els"o sz:ovegegys'egszint
c'ime nyomtat'odik ki a v'izszintes vonal al'a.

((-cpl = sz'am
(((Az a legm'elyebb sz:ovegegys'egszint, amelynek c'ime m'eg
beker:uljon a tartalomjegyz'ekbe.
Alap'ertelmez'esben a sz:ovegegys'egek
c'imei mind beker:ulnek a tartalomjegyz'ekbe.

((-rci = sz'am
(((Az a legm'elyebb sz:ovegegys'egszint, amelynek c'ime m'eg
beker:uljon az indexbe.
Alap'ertelmez'esben a sz:ovegegys'egek
c'imei nem ker:ulnek be az indexbe.

((-spn
(((Kulcsszavas param'eter, a lapsz'amoz'ast a fejezetsz'am --
azon bel:ul folyamatos lapsz'amr'ol (5.27 az :ot:odik fejezet
huszonhetedik lapja) 'at'all'itja
folyamatos lapsz'amoz'asra (113 a dokument'aci'o 113. lapja).
Ez automatikusan letiltja a lapalji
al'ah'uzott sort 'es a l'absz:oveget, valamint a
tartalomjegyz'ek nyomtat'as'at.
Ez az un. "lev'el t'ipus'u" lapsz'amoz'as.

((-npr
(((Kulcsszavas param'eter, a lapsz'amoz'as letilt'as'ara
szolg'al.

((-pag=<sz'am>
(((A lapsz'amoz'as kezd"o'ert'ek'et lehet megadni

\$pag/3.
((-cpt=<sz'am>
(((A fejezet kezd"o'ertek'enek megad'as'ara szolg'al.
El"ofordulhat, hogy egy k'esz dokument'aci'onak egy r'esz'et meg
kell v'altoztatni.
K'enyelmetlen lenne, ha ebben az esetben a
teljes dokument'aci'ot 'ujra k'ene nyomtatni.
Lehet"oseg van
arra, hogy csak a megv'altoztatott fejezetet nyomtassuk 'ujra,
megadva ezzel a param'eterrel a kezd"o fejezetsz'amot.
\$cpt.
Sz:oveg k:ozbeni vez'erl"o parancsok
Az input sz:oveg k:ozben kiadhat'o szerkeszt'esi parancsok a
k:ovetkez"ok:


```
((($pag(-(lapv'alt'as
(($spa(-(ures sor
(($tab(-(tabul'atorpoz'ici'ok 'all'it'asa
$sec.
Lapv'alt'as
A parancs form'atuma:
```

```
((($pag/[<sz'am>] [ODD] [EVE]
```

A lapv'alt'as lehet direkt 'es felt'eteles.

Ha csak a "\$pag." van

megadva, ez a felt'etel n'elk:uli lapv'alt'ast jelenti, azaz ugr'as a k:ovetkez"o lap tetej'ere.

Ha meg van adva az ODD

param'eter (\$pag/odd.), akkor a k:ovetkez"o p'aratlan lapon, ha az EVE param'eter, akkor a k:ovetkez"o p'aros lapon folytat'odik a nyomtat'as.

Ha a <sz'am> param'eter van megadva, csak akkor

t:ort'enik lapemel'es, ha a <sz'am> darab sor m'ar ezen a lapon nem f'er el.

(A h'arom param'eter k:olcs:on:osen kiz'arja egym'ast, mindig csak az egyik szerepelhet.

(A lapemel'es hat'as'ara ki'ir'odik a l'absor, 'es a k:ovetkez"o lap tetej'en az aktu'alis lapsorsz'am.

\$sec.

:Ures sor megad'asa

Az :ures sort h'arom m'odon lehet megadni.

((-(a dokument'aci'o sz:oveg'eben g'epel'eskor :ures sorokat hagyunk;

((-(az input sor csak egy "\$" karaktert tartalmaz;

((-((\$spa/<sz'am> parancssal.

((Ez a parancs lehet"os'eget ad arra, hogy egy vagy t:obb :ures sort adjunk meg a dokumentum sz:oveg'eben.

A <sz'am> param'eter

adja meg az :ures sorok sz'am'at.

Ha nincs megadva, az

alap'ertelmez'es 1.

Az els"o k'et m'odon megadott :ures sor -- ha nem f'er el az aktu'alis lapon -- lapv'alt'as ut'an az 'uj lapon nyomtat'odik.

Ha a "\$spa" paranccsal k'ert:uk az :ures sor nyomtat'as'at, akkor a felesleges :ures sorok az 'uj lapon nem nyomtat'odnak ki.

Pl.:

a "\$spa/5." paranccsal 5 :ures sort k'ert:unk, de a lapon m'ar csak 3 sor f'er el, a lapemel'es ut'an nem nyomtat'odik ki a 2 hi'anyz'o sor.

Viszont ha 5 :ures sort g'epelt:unk, vagy 5 csak

"\$"-t tartalmaz'o sort, akkor a fennmarad'o 2 sor a k:ovetkez'o lapon :ures sork'ent megjelenik.

\$pag/5.

\$sec.

Tabul'atorpoz'ici'ok 'all'it'asa

A DOG :osszesen 20 tabul'atorpoz'ici'ot tud 'ertelmezni.

(Alap'ertelmez'esben az:

(1, 4, 8, 12, 16, 68, 72, 76 sorozat 'erv'enyos.

A tabul'atorpoz'ici'ok az input sz:ovegben b'armikor 'atdefini'alhat'ok a "\$tab" paranccsal.

A parancs form'atuma:

(((\$tab/x,x,x,x,x, ,x,x,x.

Ahol x az a poz'ici'o ahov'a a tabul'atort szeretn'enk 'all'itani.

A nem 'all'itott tabul'ator\poz'ici'ok 'ert'eke 0.

A tabul'atorpoz'ici'ok

mindig a bal marg'ot'ol sz'am'it'odnak.

Ha a

tabul'ator\'all'it'as kil'og a sorb'ol, a DOG hibajelz'est 'ir ki, 'es a

poz'ici'ot figyelmen k'iv:ul hagyja.

A tabul'ator\ugr'as jele az input sz:ovegben a nyit'o kapcsos z'ar'ojel.

A t'engleges tabul'ator\karakterek csak a "k'ep"

parancsokon bel:ul haszn'alhat'ok, egy'ebk'ent sz'ok:ozzel helyettes'it'odnek.

\$cpt.

Egy'eb parancsok.

\$sec.

A dokumentum c'imoldal'ának a megad'asa (\$tit)
Ezzel a paranccsal lehet a dokument'aci'o c'imoldal'at megadni.
A c'imoldal a t'enyleges sz:oveg el"ott szerepel,
teh'at nincs lapsz'amo'asa (a lapsz'amo'as a c'imoldal v'ege
ut'an kezd"odik el).

A parancs szintaxisa:
{ \$tit. [<a dokument'aci'o neve>]

A c'implap ilyen m'odon teh'at lapsz'amo'as n'elk:uli
sz:ovegr'esz.

A \$tit parancs a dokument'aci'o nev'et is megadja.
Ezt a nevet haszn'aljuk
arra, hogy a p'aros (bal oldali) lapokat azonos'itsuk - l'asd a
lapsz'amo'assal foglalkoz'o fejezetet.
Ezt a nevet a \$tit parancsban l'ev"o
"." karakter ut'an adjuk meg.
Ha a parancs szerepel a forr'asban, akkor meg kell, hogy el"ozze
az els"o kinyomtatand'o sz:ovegr'eszt.

\$sec.

A szabv'anyos nevek megad'asa (\$sum, \$reg)
Ezek a parancsok lehet"ov'e teszik, hogy a dokument'aci'o
'alland'o r'eszeinek elnevez'eseit megadjuk.

A n'ev megad'asa t'enylegesen a parancs ut'an
t:ort'enik (a parancs z'ar'okaraktere a "." ut'an).
A tartalomjegyz'ek 'es index nev'et a
dokumentum megfelel"o r'esz'eben fejl'ecfelirat'ent haszn'aljuk,
de azonos'it'o sz:ovegk'ent is kinyomtatjuk a megfelel"o
lapok alj'ara vagy tetej'ere (ld. lapoz'as r'eszt).

\$pag/5.

A parancsok szintaxisa:

{ \$sum [/ <a-lapok-sz'amo'asa>]. <a-tartalomjegyz'ek-neve>

{ \$reg [/ <a-lapok-sz'amo'asa>]. <az-index-neve>

\$pag/5.

A "\$sum" a tartalomjegyz'ek nev'et adja meg, a "\$rem" pedig az
index'et. A \$rem parancs egyuttal el"oirja az index

l'etrehoz'as'at 'es nyomtat'as'at is.

Az els"o param'eter egy bet"u,
amit a lapok sz'amoza's'an'al haszn'alunk (ld. a lapoz'asn'al).
Alap'ertelmez'es szerinti 'ert'eke tartalomjegyz'ekn'el a
"T" bet"u, indexn'el az "I". A m'asodik param'eter az index
illetve a tartalomjegyz'ek c'ime. Az indexn'el
alap'ertelmez'esben I^N^D^E^X, a tartalomjegyz'ekn'el pedig
T^A^R^T^A^L^O^M.

\$sec.

Be'agyazott input file megad'asa (\$inp)

Ezzel a paranccsal m'asolhat'o egy adott input file-ba egy
m'asik file.

A parancs form'atuma:

(((\$inp/<file-n'ev>

ahol a <file-n'ev> az MS-DOS 'altal megengedett
n'evkonveci'oknak kell, hogy megfeleljen.

\$sec.

A sz:oveges lapok azonos'it'asa

A dokument'aci'o minden sz'amozott lapja tartalmazhat
olyan sz:oveget is a lapokon, ami a lapot azonos'itja.

Form'aj'anak 'es

haszn'alat'anak szab'alyai a k:ovetkez"ok:

((-Ca azonos'it'o sz:oveg nyomtathat'o:

(((((*(a lap alj'ara, azaz az adott lapon l'ev"o sz:oveg al'a
(l'absor);

(((((*(a lap tetej'ere, azaz az adott lapon l'ev"o sz:oveg
f:ol'e ('el"ofej);

((-Ca kinyomtat'as letilthat'o a "\$par" parancs "noi"
param'eter'enek seg'its'eg'evel;

((-Ca sz:oveg kinyomtat'asa el"ott :ures sorok maradnak ki,
ezeknek a

sz'ama megadhat'o a "sap" param'eterrel a \$par parancsban
(alap'ertelmez'es szerint 1 :ures sor marad ki);

\$pag/4.

((-Ca lapot azonos'it'o sz:oveg el"ott vagy ut'an egy "--

karakterekb"ol 'all'o
vizzszintes vonal is kinyomtat'asra ker:ul, de ha feleslegesnek
tal'aljuk,
akkor ez megsz:untethet"o a "nou" param'eter megad'as'aval az
eml'itett helyen;

{{-{{az azonos'it'o sz:ovegre vonatkoz'o szab'alyok szerint
azonos'it'asra
a k:ovetkez"o sz:ovegek (nevek) valamelyike haszn'alhat'o:

{{{{*{{a "\$tit" parancssal legut'obb megadott dokument'aci'on'ev
a dokument'aci'o minden r'esz'en'el a bal oldali, p'aros lapok
alj'ara fog ker:ulni;

{{{{*{{a jobb oldali, p'aratlan lapok alj'ara a fejezet c'ime fog
ker:ulni, ez att'ol f:ugg, hogy a dokumentum mely r'esz'er"ol
van sz'o:

{{{{.{{a sz:oveges r'eszn'el ez a legut'obb megadott fejezet
neve lesz, azaz a "\$cpt" parancsn'al megadott sz:oveg,
vagy a fejezet nev'enek els"o sora, felt'eve, hogy ez a sz:oveg
nem lett 'atdefini'alva a "\$cpt" parancsban a "." karakter
ut'an;

{{{{.{{a tartalomjegyz'ek eset'en ez a
megfelel"o r'esz neve, amit a legut'obbi "\$sum"
(tartalomjegyz'ek) parancs seg'its'eg'evel adtunk
meg (ha ilyen parancsot nem haszn'altunk, akkor a
standard sz:oveg lesz 'erv'enyos -- T^A^R^T^A^L^O^M).
\$sec.

'El"ofej nyomtat'as'anak k'er'ese
A lapazonos'it'asi lehet"os'egek egyike.

Haszn'alata kiz'arja a
l'absor nyomtat'as'at.

Az 'el"ofej azt jelenti, hogy a lap tetej'en ki'ir'odik
egy azonos'it'o sz:oveg 'es a lapsz'am.

Ez v'izzszintes vonallal al'a lesz h'uzva.

A lapsz'am mindig a lap k:uls"o

sz'el'en jelenik meg, teh'at a bal oldali lapokon a sz:oveg
el"ott, a jobboldaliakon pedig a sz:oveg ut'an.

A parancs form'aja:

(\$LHD.

\$cpt.

A tartalomjegyz'ek l'etrehoz'asa 'es nyomtat'asa (\$con)

A tartalomjegyz'ek olyan sz:oveg\egys'eg\nevek halmaz'a,

amelyeket a "\$cpt", "\$sec", "\$prg", "\$art"

'es "\$itm" parancsokkal adtunk meg.

A tartalomjegyz'ek l'etrehoz'as'anak

'es nyomtat'as'anak a k:ovetkez'o igen egyszer:u szab'alyai

vannak:

((-(a "cpl" param'eter ("par" parancs) korl'atozhatja a

sz:oveg\egys'eg\nevek

tartalomjegyz'ekbe v'etel'et - a sz:oveg\egys'eg\nevek az

1 szint'u fejezett"ol kezdve a megadott szintig ker:ulnek

a tartalomjegyz'ekbe (ennek implicit 'ert'eke 5, ekkor az

:osszes sz:ovegegys'egn'ev beleker:ul a tartalomjegyz'ekbe);

((-(csak azok a sz:oveg\egys'eg\nevek ker:ulnek a

tartalomjegyz'ekbe, amelyek a dokument'aci'o sz:oveg'eben

el"ofordulnak;

((-(a k:ovetkez'o nyilv'anval'o t'eny az, hogy a

sz:ovegegys'egek a megad'asuk sorrendj'eben r:ogz'it"odnek,

'es a tartalomjegyz'ek ilyen m'odon

a lapsz'amok szerint rendezett lesz;

((-(a tartalomjegyz'ek l'etrehoz'asa (azaz a t'etelek

- sz:oveg\egys'eg\nevek

bejegyz'ese a tartalomjegyz'ekbe) minden esetben megt:ort'enik

(m'eg akkor is, ha a "\$con" funkci'ot nem k'ert:uk).

A "\$con" parancs teh'at

mind:ossze a tartalomjegyz'ek nyomtat'as'at k'eri.

Ha a

parancsot nem adjuk meg, a tartalomjegyz'ek nem 'ir'odik ki;

((-(a sz:ovegegys'egn'ev form'atuma ugyanaz, mint ahogy a

sz:ovegben szerepel (m'eg a sz:oveg\egys'egsz'am eltol'asa

is megegyezik), az

automatikus al'ah'uz'as term'eszetesen nem fog megjelenni;

{{-{{a \$cpt-el megadott sz:oveg\egys'eg nevet az azt megel"oz"o sz:oveg\egys'eg\n'evt"ol egy :ures sor v'alasztja el;

{{-{{a n'ev utols'o kifejez'es'et egy sz'ok:oz k:oveti 'es ezut'an pontozott vonal ker:ul ("." karakterekb"ol - pontokb"ol), ami eg'eszen a lap jobb oldal'aig tart (elmaradhat, ha hossz'u a sz:oveg\egys'eg neve), ezt k:oveti az a lapsz'am, ahol a sz:oveg\egys'eg megtal'alhat'o.

A lapsz'amok a lap jobb sz'el'ere igaz'it'odnak.

\$cpt.

Az indext'etelek megad'asa 'es nyomtat'asa (\$rg)

Az indext'etelek azonos'it'asa k'etfelek'eppen t:ort'enhet:

{{-{{olyan sz:oveg\egys'eg\nevek ker:ulhetnek be, amelyeket a "\$cpt", "\$sec", "\$prg", "\$art" 'es "\$itm" parancsokkal adtunk meg.

Az "rcl" param'eter ("\$par" parancs) el"o'irhatja a sz:oveg\egys'eg\nevek

indexbe v'etel'et -- a sz:oveg\egys'eg\nevek alap'ertel'mez'esben nem ker:ulnek be az indexbe.

{{-{{a \$rg. parancssal azonos'itott sorok tartalma minden esetben indext'etel.

{{{{A parancs form'aja:

{{{{{{(\$rg.<sz:oveg>

{{{{{{ahol a <sz:oveg> param'eter az indext'etel sz:ovege. Ez a parancs nem szak'itja meg az input folyamatos feldolgoz'as'at, b'ar "\$" karakterrel kezd"odik, a sorok t:ordel'es'en'el nem funkcion'al parancsk'ent.

Az index l'etrehoz'asa (azaz a t'etelek bejegyz'ese az indexbe) csak abban az esetben t:ort'enik meg, ha a "\$reg" funkci'ot k'ert:uk. Ez egyben el"o'irja a nyomtat'ast is.

Az indext'etelek a k:ovetkez"o form'aban nyomtat'odnak:

{{-{{n'evsorba rendezetten (a n'evel"ok -- a, az -- figyelembe v'etele n'elk:ul;

{{-Ca t'etel m:og:ott az oldalsz'amok, ahol a t'etelek
azonos'it'asa t:ort'ent;

{{-Cazonos t'etelek egyszer nyomtat'odnak, t:obb oldalsz'ammal.
\$cpt.

Bet"ut'ipus 'es soremel'es 'all'it'asa, :uzenatk:uld'es konzolra
\$sec.

Bet"ut'ipus 'all'it'asa

Az input sorban megadott \$eli parancs "elite size" m'odra,
a \$nor parancs "pica size" (norm'al) m'odra, a \$cnd
condensed (:összeh'uzott) m'odra 'all'itja a sornyomtat'ot.
Ez a m'od marad mindaddig,
am'ig 'at nem 'all'itjuk m'as m'odra.

\$sec.

Soremel'es 'all'it'asa

Lehet"os'eg van minden egyes soremel'es dupla soremel'ess'e
alak'it'as'ara a \$dou paranccsal.

Mindaddig, am'ig a \$nod

parancsot ki nem adjuk, minden sor ut'an egy :ures sor
nyomtat'odik.

\$sec.

Feldolgoz'as k:ozbeni :uzenatk:uld'es

A sz:oveg feldolgoz'asa k:ozben lehet"os'eg:unk van :uzenet
k:uld'es'ere.

{A parancs form'aja:

(\$rem/pause.<sz:oveg>

A parancs a <sz:oveg>-ben megadott karaktersorozatot a
k'eperny"or'e nyomtatja. Ha haszn'altuk a "pause" param'etert,
akkor

a feldolgoz'as mindaddig nem folytat'odik, m'ig a

<sz:oveg> megjelen'es'et k:ovet"oen egy

billenty"ut le nem :ut:unk. Ha nem haszn'altuk, csak a <sz:oveg>
'ir'odik ki, a feldolgoz'as nem szakad meg.

\$cpt.

A DLOOK, CONVERT 'es CEDIT seg'edprogramok haszn'alata

\$sec.

A DLOOK seg'edprogram

A DLOOK seg'edprogram lehet"os'eget ad a megszerkesztett
dokument'aci'o nyomtat'as 'el"otti megtekint'es'ere a
k'eperny"on.

A DOG (ha nem adunk meg output file-t) l'etrehoz egy <input file>.DTF nev"u k:oztes file-t, amelyet inputk'ent megadva a DLOOK programnak, a sz:oveg szerkesztett form'aban megjelenik a k'eperny"on. K:ul:onb:oz"o sz'inekkal jal:oli a kifejez'es kiemel'eseket. Piros, ha al'ah'uzott, k'ek, ha d"olt bet"us, intenz'iv, ha k:ov'er'itett a kifejez'es. A DLOOK help-je a "?" megnyom'as'ara aktiviz'al'odik. A file beolvas'asa ut'an a kezel'es legegyszer"ubben a kurzormozgat'o billenty"ukkel t:ort'enhet.

Ind'it'asa:

>DLOOK <inputfile>

\$sec.

A CEDIT 'es CONVERT seg'edprogramok

Ez a programp'aros lehet"os'eget ny'ujt arra, hogy a DOG 'altal el"o'all'itott k:oztes file-b'ol nyomtathat'o file-t 'all'itsunk el"o. Erre az'ert van sz:uks'eg, hogy b'armely nyomtat'o haszn'alhat'o legyen a dokument'aci'o ki'irat'as'ara.

A CEDIT programmal

el"o kell 'all'itani az 'atk'odol'o file-t, melynek felhaszn'al'as'aval a CONVERT program elv'egzi a file 'atk'odol'as'at. Mindk'et program konvert'al'o file-okkal dolgozik. A konvert'al'o file-okat c'elszer"u egy k:onyvt'arban tartani. Így viszont k'enyelmetlen, hogy h'iv'akor mind'ig a teljes (k:onyvt'arn'evvel megadott) file-nevet kell be:utni. Ez'ert a programok el"osz:or az aktu'alis k:onyvt'arban, majd ha ott nem tal'alt'ak a rendszer SET parancs'aval megadott DOGLIB k:onyvt'arban keresik a h'ivott konvert'al'o file-t. A SET parancsot 'erdemes az AUTOEXEC.BAT file-ba beletenni a k:ovetekez"o form'aban:

((set DOGLIB=<k:onyvt'arn'ev>

ahol a <k:onyvt'arn'ev> nev"u k:onyvt'arban vannak a konvert'al'o file-ok.

\$pag/10.

\$prg.

A CEDIT program haszn'alata

A program feladata egy olyan konvert'al'o file l'etrehoz'asa, melynek haszn'alat'aval a CONVERT program az 'atalak'it'asokat elv'egezheti.

Teh'at itt a CEDIT programban kell megadnunk, hogy
a DOG outputfile-j'aban l'ev"o 'ekezetes 'es vez'erl"o
karakterek milyen karakternek felelnek meg a nyomtat'on.

A program k'et k'eperny"ovel dolgozik:

{{-{{az els"o k'eperny"on adhatjuk meg a karakterk'eszlet egy
elem'enek cser'ej'et egy m'asik elemre.

{{-{{a m'asodik k'eperny"on jel:olhetjuk ki b'armely
karaktersorozat b'armilyen m'asik karaktersorozatra val'o
cser'ej'et.

A program elind'ithat'o input file megad'as'aval 'es an'elk:ul.
Az els"o esetben egy m'ar megl'ev"o 'atk'odol'o file-t akarunk
jav'itani, a m'asodik esetben pedig egy 'ujat hozunk l'etre.

A k:ovetkez"o t'abl'azatban p'eldak'ent le'irjuk, hogy melyik
karaktereket
mire kell cser'elni a k'et m'atrixnyomtat'o 'es a k'eperny"o
eset'en.

Az els"o oszlopban megadjuk a bet"ut, a m'asodikban a DOG
k:oztesfile-j'aban lev"o 'ert'eket, a harmadikban a k'eperny"ore
'irat'asn'al haszn'alhat'o 'ert'eket decim'alisan.

A negyedik
oszlop a

VIDEOTON gy'artm'any'u nyomtat'ok, az :ot:odik pedig az EPSON RX
t'ipus'u nyomtat'ok k'odjait tartalmazza.

\$pag/40.

\$pic.

```
+=====+
| Bet"uk | DOG |Consol |VT21200| Epson RX |
+=====+
| 'a | 130 | 160 | 224 | ESC,"R",6,"C",ESC,"R",0 |
+-----+
| 'A | 131 | 143 | 192 | "A", 8,"" |
+-----+
| 'e | 132 | 130 | 225 | ESC "R",2,"C",ESC,"R",0 |
+-----+
| 'E | 133 | 144 | 193 | ESC "R",5,"@",ESC,"R",0 |
+-----+
| 'i | 134 | 161 | 226 | "i", 8,"" |
+-----+
| 'I | 135 | 73 | 73 | "I", 8,"" |
+-----+
| 'o | 136 | 162 | 227 | ESC "R",6,"I",ESC,"R",0 |
+-----+
| 'O | 137 | 162 | 195 | ""O", 8,"" |
+-----+
| ::o | 138 | 148 | 228 | ESC "R",5,"I",ESC,"R",0 |
+-----+
| ::O | 139 | 153 | 196 | ESC "R",5,"\",ESC,"R",0 |
+-----+
| ""o | 140 | 148 | 229 | ""o", 8,"" |
+-----+
| ""O | 141 | 153 | 197 | ""O", 8,"" |
+-----+
| 'u | 142 | 163 | 230 | ""u", 8,"" |
+-----+
| 'U | 143 | 163 | 198 | ""U", 8,"" |
+-----+
| ::u | 144 | 129 | 231 | ESC "R",2,">",ESC,"R",0 |
+-----+
| ::U | 145 | 154 | 199 | ESC "R",2,"J",ESC,"R",0 |
+-----+
| ""u | 146 | 129 | 232 | ""u" 8 "" |
+-----+
| ""U | 147 | 154 | 200 | ""U" 8 "" |
+-----+
$eoc.
```


\$pag/23.

\$pic/tc.

```
+-----+-----+-----+-----+
|      K'odok      | DOG |Consol |VT21200|   Epson RX   |
+-----+-----+-----+-----+
| al'ah'uz'as be   | 150 |      |ESC""U" | ESC "-" 1    |
+-----+-----+-----+-----+
| al'ah'uz'as ki   | 151 |      |ESC""U" | ESC "-" 0    |
+-----+-----+-----+-----+
| dupla nyomt. be  | 152 |      |ESC"R"  | ESC "G"      |
+-----+-----+-----+-----+
| dupla nyomt. ki  | 153 |      |ESC"R"  | ESC "H"      |
+-----+-----+-----+-----+
| norm'al bet"ut'ipus | 154 |      |ESC "-" | ESC "P" 18   |
+-----+-----+-----+-----+
| elite bet"ut'ipus | 155 |      |ESC"*"  | 18 ESC "M"   |
+-----+-----+-----+-----+
| cond. bet"ut'ipus | 156 |      |ESC"+"  | ESC "P" 15   |
+-----+-----+-----+-----+
| italic m'od be   | 157 |      | nincs | ESC 4        |
+-----+-----+-----+-----+
| italic m'od ki   | 158 |      | nincs | ESC 5        |
+-----+-----+-----+-----+
| inicializ'al'o kar.| 159 |      |      |              |
+-----+-----+-----+-----+
$eoc.
```

\$pag/7.

Az inicializ'al'o karakter arra szolg'al, hogy a nyomtat'onak minden lehet'os'eg'et kihaszn'alhassuk.

Ezt a karaktert a DOG

indul'askor azonnal kiteszi a k:oztes file-ba.

Ha a

sornyomtat'onak vez'erl'o karaktereket akarunk adni, ezt a karaktert cser'eljuk a konvert'al'as sor'an a vez'erl'o sorozatra.

Ha nem akarjuk felhaszn'alni semmire, a konvert'al'o tabl'aban egy nulla hossz'us'ag'u stringre v'altsuk ki.

\$prg.

A CONVERT program

Ezzel a programmal v'egezhetjuk el az output file v'egleges form'ara hoz'asat.

Minden karaktert vagy karaktersorozatot a
CEDIT programmal el"o'all'itott konverter file alapj'an
a megadottra cser'el.
\$pag/5.
A CONVERT program h'iv'asa:

```
{{convert input_file output_file konverter_file
```

Az input file, melyet a DOG programmal 'all'it'ottunk el"o,
alap'ertelmezett kiterjeszt'ese ".DTF".

Az output file-t a felhaszn'al'o k:oteles
megadni, a rendszer file-n'ev konvenci'oinak megfelel"oen.
Ha a file-t'ipust

nem adja meg, akkor az

alap'ertelmez'es az "input_file.DOC" file-n'ev.

A

konverter file-t a CEDIT seg'edprogrammal kell el"o'all'itani,
alap'ertelmezett n'evkiterjeszt'ese ".CNV".

\$cpt.

A DOG haszn'alata

A DOG egy sz:ovegfile-t dolgoz fel.

Ezt a file-t

el"o'all'ithatjuk b'armilyen editorral, amelyik nem tesz
speci'alis karaktereket a file-ba, mi a szerkeszt'es folyam'an
a NORTON editort 'es a Perfect Writer-t haszn'altuk a sz:ovegek
beg'epel'es'ere.

A sz:oveg elk'esz:ulte ut'an meg kell h'ivni a

DOG-ot a k:ovetkezo m'odon:

```
>DOG input [opci'ok]
```

Opci'ok:

```
{{(-o <fn'ev>): az output file
```

```
{{(-c <fn'ev>): a konvert'al'o file neve.
```

Ha nem adunk meg

konvert'al'o file-t, akkor a CONVERT seg'edprogram nem h'iv'odik
meg, s az output file megtekinthet"o a DLOOK programmal.

A "-c"

opci'o haszn'alata csak teljesen k'esz dokument'aci'on'al
c'elszer"u.

Ha nem adjuk meg param'eterk'ent, a CONVERT
:on'all'oan is megh'ivhat'o a DOG output file-j'ara.

```
(((-t<bet"u> a karakter t'ipusa
(((Ch a bet"u:
((((e(-<Elite size
((((n(-<norm'al
((((c(-<condensed
```

```
(Alap'ertelmezett input n'evkiterjeszt'es: .TXT
(Alap'ertelmezett output: <inputfile>.DTF
(Alap'ertelmezett karaktert'ipus: norm'al
```

Ha a DOG-ot nem -c opci'oval h'ivtuk meg, akkor ezut'an a
CONVERT programot kell r'aengedni az el"o'all'itott file-ra.
\$cpt.

A parancsok r:ovid ismertet'ese
Minden parancs alatt megadjuk azt az angol sz'ot, melyb"ol a
r:ovid'it'es sz:uletett.

```
$pag/5.
($art[/<sz'am>].
((art -> article
```

```
((Negyedik szint"u sz:ovegegys'eg.
```

```
$pag/5.
($con.
((con -> content
```

```
((A tartalomjegyz'ek nyomtat'as'anak k'er'ese.
```

```
$pag/5.
($cnd.
((cnd -> condensed
```

```
((Condensed bet"ut'ipus'u nyomtat'as.
```

```
$pag/5.
($cpt[/<sz'am>].<sz:oveg>]
((cpt -> chapter
```


((Els"o szint"u sz:ovegegys'eg.

\$pag/8.

(\$ctr[/**lnb**=]**lapm'eret**],[**cnb**=]**sorm'eret**],
[**mar**=]**bal^marg'o^poz.**),
tab=**tabu\l'a\tor\ka\rak\ter**),
exc=**kifejezes osszekoto karakter**).
((ctr -> control
((Lapm'eret defini'al'asa.

\$pag/3.

(((((**lnb**=**lapm'eret**)
(((((((**lnb** -> line number
(((((((a lapon elhelyezend"o sorok sz'ama

\$pag/3.

(((((**cnb**=**sorm'eret**)
(((((((**cnb** -> character number
(((((((az egy sorba ker:ul"o karakterek sz'ama

\$pag/3.

(((((**mar**=**bal marg'o pozici'o**)
(((((((**mar** -> margin
(((((((a lap bal sz'els"o karakterpozici'oj'anak be'all'it'asa

\$pag/3.

(((((**tab**=**tabu\l'a\tor\ka\rak\ter**)
(((((((**tab** -> tabulator

\$pag/3.

(((((**exc**=**kifejez'es:osszek:ot"o karakter**)
(((((((**exc** -> expression continue

\$pag/5.

(\$dou.
((dou -> double

((Dupla soresmel'es bekapcsol'asa.

\$pag/5.

(\$eli.
((eli -> elite

((Elite bet"ut'ipus'u nyomtat'as.

\$pag/5.

(\$eoc.

((eoc -> end of command

((K:ul:onleges feldolgoz'ast lez'ar'o parancs.

\$pag/5.

(\$inp/<file-n'ev).

((inp -> input

((Be'agyazott file kijel:ol'ese.

\$pag/5.

(\$itm[/<sz'am)].

((itm -> item

((:Ot:odik szint"u sz:ovegegys'eg.

\$pag/5.

(\$mid[/tc].

((mid -> middle

((K:oz'epre helyezend"o k'ep kezdete.

\$pag/5.

(\$nod.

((nod -> no double

((Dupla soremel'es kikapcsol'asa.

\$pag/5.

(\$nor.

((nor -> normal

((Norm'al (Pica size) bet"ut'ipus'u nyomtat'as.

\$pag/5.

(\$pic[/tc].

((pic -> picture

{{K'ep kezdet'et defini'al'o parancs.

\$pag/5.

{{prg[/<sz'am>].

{{prg -> paragraph

{{(Harmadik szint"u sz:ovegegys'eg.

\$pag/5.

{{pag[/<sz'am> ODD EVE].

{{pag -> page

{{(Lapv'alt'as.

\$pag/10.

{{par{[/[pos=]<a sz:ovegegys'eg c'im'enek elhelyez'ese>,

{{{{{{[sbc=]<:ures sorok sz'ama a sz:ovegegys'eg c'ime el"ott>,

{{{{{{[sac=]<:ures sorok sz'ama a sz:ovegegys'eg c'ime ut'an>,

{{{{{{[for=]<a c'im 'ir'asi m'odja>,

{{{{{{[sbp=]<a lapsz'amot k:ovet"o :ures sorok sz'ama>,

{{{{{{[sap=]<a l'absort megel"oz"o :ures sorok sz'ama>,

{{{{{{[nou], [noi], [cpl], [rc], [spn], [nnp], [pag=<sz'am>],

[cpt=<sz'am>].

{{par -> parameters

{{(A lap 'es sz:ovegegys'eg form'atum'anak be'all'it'asa.

{{{{{{{{(pos = a sz:ovegegys'eg c'im'enek elhelyez'ese

{{{{{{{{(pos -> positions

{{{{{{{{(A lehets'eges 'ert'ekek:

\$pag/3.

{{{{{{{{{{(op: (A sz:ovegegys'eg a k:ovetkez"o p'aratlan sz'am'u
lap tetej'en kezd"odik (odd page).

\$pag/3.

{{{{{{{{{{(ep: (A sz:ovegegys'eg a k:ovetkez"o p'aros sz'am'u
lap tetej'en kezd"odik (even page).

\$pag/3.

{{{{{{{{{{(np: (A sz:ovegegys'eg a k:ovetkez"o lap tetej'en
kezd"odik (next page).

\$pag/3.

{{{{{{{{{{(no: (A sz:ovegegys'eg nem ker:ul 'uj lapra (no page).

((((((sbc = :ures sorok sz'ama a sz:ovegegys'eg c'ime el"ott
 ((((((sbc -> space before chapter

 ((((((sac = :ures sorok sz'ama a sz:ovegegys'eg c'ime ut'an
 ((((((sac -> space after chapter

 ((((((for = a c'im 'ir'asi m'odja
 ((((((for -> format
 ((((((Lehets'eges 'ert'ekei:
 (((((((no)a c'im al'ah'uz'as n'elk:ul lesz ki'irva (not)
 (((((((un)a c'im al'ah'uzva lesz ki'irva (underline)

 ((((((sbc = a lapsz'amot k:ovet"o :ures sorok sz'ama
 ((((((sbc -> space before page

 ((((((sap = a l'absort megel"oz"o :ures sorok sz'ama
 ((((((sap -> space after page

 ((((((nou = a lapok alj'ara vagy tetej'ere nem kell v'izszintes
 vonalat h'uzni
 ((((((nou -> no underline

 ((((((noi = nem kell l'absort vagy 'el"ofejtet nyomtatni
 ((((((noi -> no identify

 ((((((cpl = tartalomjegyz'ekbe ker:ul"o legkisebb
 sz:ovegegys'eg m'elys'ege
 ((((((cpl -> chapter level

 ((((((rcl = indexbe ker:ul"o legkisebb
 sz:ovegegys'eg m'elys'ege
 ((((((rcl -> register count level

 ((((((spn = folyamatos lapsz'amoz'as
 ((((((spn -> single page number

 ((((((nnp = lapsz'amoz'as letilt'asa
 ((((((no page number

 ((((((pag = lapsz'amoz'as kezd"o 'ert'eke
 ((((((pag -> page


```

$pag/2.
(((((-{cpt = fejezetsz'am kezd'o 'ert'eke
(((((((cpt -> chapter

$pag/5.
($reg/rom c.[<sz:oveg>]
((reg -> register

(((Az index lapsz'amoz'asa.

$pag/5.
($rem/pause.
((rem -> remark

((Menetk:ozbeni :uzenetk:uld'es a k'eperny"ore.

$pag/5.
($rg.
((rg -> register

(((Indext'etel azonos'it'asa.

$pag/5.
($spa[/<sz'am>].
((spa -> space

(((:Ures sor megad'asa.

$pag/5.
($sec[/<sz'am>].
((sec -> section

((M'asodik szint"u sz:ovegegys'eg.

$pag/5.
($shd.
((shd -> set header

((A t'abl'azat fejl'ec'enek a megad'asa.

$pag/10.
($stb[/<[num=]<sorok>,<vch=]<f:ugg"oleges-vonal>,<?

```


(((hch=<vizszintes-vonal>,[cch=<metsz'espontok>)?
((tc,unl,spa,nspv,nsph].
((stb -> set table

(((A t'abl'azat kezd"o parancsa.

(((-(num(((hogya h'any sor ut'an kell a t'abl'azatban egy
v'izszintes elv'alaszt'ovonalat automatikusan beilleszteni
(number).

\$pag/2.

(((-(spa(((az elv'alaszt'o sor egy :ures sor legyen (space).

\$pag/3.

(((-(unl(((az elv'alaszt'o sor egy v'izszintes vonal legyen
(underline).

\$pag/3.

(((-(vch(((a f:ugg"oleges vonal karaktere (vertical character).

\$pag/3.

(((-(hch(((a v'izszintes vonal karaktere (horizontal
character).

\$pag/3.

(((-(cch(((a metsz'espontokban haszn'alt karakter
(cross character).

\$pag/3.

(((-(nspv(((a t'abl'azat f:ugg"oleges vonalai el"ott
'es ut'an k:ozvetlen:ul ne maradjon ki :ures hely
(no space vertical).

\$pag/3.

(((-(nsph(((a vizszintes vonalak el'e vagy ut'an ne
ker:uljon :ures sor
(no space horizontal).

\$pag/3.

(((-(tc(((a t'etelek (kifejez'esek) 'atk'odol'asra ker:ulnek.
(text convert).


```

$pag/5.
($sum/rom c.[<sz:oveg>]
{($sum -> summary

{{{A tartalomjegyz'ek lapsz'amosz'asa.

$pag/5.
($unl[/<sz'am> spa].
{($unl -> underline

{{{V'izszintes vonal h'uz'asa a t'abl'azatban.

$pag/5.
($vel/<f:ugg"oleges vonalpoz'ici'ok>/<t'etelek elhelyez'ese>.
{($vel -> vertical lines

{{{A f:ugg"oleges vonalak poz'ici'oj'anak 'es a
t'abl'azatt'etelek elhelyez'es'enek megad'asa.

$pag/5.
($tab/x,x,x,x, .....,x,x,x.
{($tab -> tabulator
{{{Tabul'atorpoz'ici'ok 'all'it'asa.

$pag/5.
($tit.<sz:oveg>
{($tit -> title

{{{Fejl'ec vagy l'absor kijel:ol'ese.

$inp/pelda.txt

```


AKADÉMIAI KIADÓ ÉS NYOMDA

Felelős kiadó: Hazai György főigazgató

Felelős szerkesztő: Czere Károlyné

A borítót Szabadi Zoltán tervezte

ISBN 963 05 4742 2

A kézirat lezárva: 1987. szeptember

OPTIMUM – MGKSZ NY. 87. 352.

DOG -- mint Dokumentáció Generátor

"Bár egy külső megbízás miatt segédeszközként készült el a DOG őse, saját használatunkban, saját igényeink alapján fejlődött ki azzá, ami" -- mondják a szerzők Schill Róbert és Szegedi János, a Közgazdasági Egyetem fiatal munkatársai.

- Mit tud a DOG?
- Tudja mindazt, amit általában a szövegformázók: sorkiegyenlítést, lapszámozást, tartalomjegyzéket stb., és természetesen a magyar ékezeteket.
- Tud-e többet, mint a szövegformázók?
- Kényelmes táblázatelőállító parancsokat, "élőfejet" vagy "lábsort", tárgymutatót és például máshonnan beszerezhető programlistákat, tetszőleges karakterkonverziót.
- Kiknek ajánlható?
- Akik hosszabb - rövidebb tanulmányokat, dolgozatokat, dokumentációkat készítenek.

Sorozatunk minden anyagát a DOG segítségével szerkesztettük és nyomtattuk ki.

Fogy. ár: 8000 Ft.

