

MTA Számítástechnikai és Automatizálási Kutató Intézet Budapest



MAGYAR TUDOMÁNYOS AKADÉMIA
SZÁMITÁSTECHNIKAI ÉS AUTOMATIZÁLÁSI KUTATÓ INTÉZET



LIDI-72

Listakezelő rendszer a Digitális osztályon
1972 évi változat

Irta:

FIDRICH ILONA
UZSOKY MIKLÓS

Lektorálta:

MÁTÉ LEVENTE

TANULMÁNYOK 16/1974

1. BEVEZETÉS

A LIDI-72 az MTA Számítástechnikai és Automatizálási Kutató Intézete Digitális Technika Osztályán a számítógépes tervezés adatkezelő rendszereként kidolgozott rendszer 1972. évi változata.

A számítógépes tervezés e rendszerében az információ *olyan irányított gráf, amelynek bármely szögponthoz tetszésszerű számú attributum és egy név lehet rendelve*, de amelyben egy szögponthoz nem a neve, hanem a szögponthoz vezető valamely úton fekvő szögpontok neveinek sorozata azonosít. A tervezés minden szakaszában mind a – rendszerben már katalogizált vagy a felhasználó által szolgáltatott – kiindulási adatok, mind a tervezési szakasz eredményei erdőt alkotnak. Az egyes tervezési szakaszokon belül azonban nemcsak a szögpontokhoz rendelt attributumok értékei változnak, hanem módosul a gráf szerkezete is; a módosítás eredményeként előálló gráf általános szerkezetű (erősen összefüggő részgráfokat tartalmazó) gráf is lehet.

A fentiek alapján a számítógépes tervezés e rendszerében az információ egységes kezeléséhez olyan adatkezelő rendszer szükséges, amely:

- biztosítja az olyan általános szerkezetű irányított gráfok *számítógépen belüli reprezentációját*, amelyek bármely szögponthoz egy név és tetszésszerű számú attributum lehet rendelve; e belső reprezentációnak lehetővé kell tennie mind az egyes szögpontok attributumaihoz való közvetlen hozzáférést, mind a gráf szerkezetének egyszerű módosítását;
- elvégez a reprezentált irányított gráfokon definiált bizonyos *általános műveleteket*, mint pl. egyes szögpontok megkeresése, élek és szögpontok bekapcsolása, részgráfok törlése, bejárása, stb.;
- lehetővé teszi a belső reprezentációban adott gráfok átvitelét az operatív tároló és a *közvetlen hozzáférésű háttértárolók* között;
- lehetővé teszi az attributumokkal ellátott szögpontokkal rendelkező erdők általános és a tervező mérnökökhöz közelálló formában történő *leírását*;
- megoldja ilyen erdők *be- és kivitelét*;
- lehetővé teszi nagy mennyiségű, erdő szerkezetű információ *katalogizálását* a soros hozzáférésű háttértárolókon;
- biztosítja a katalogizált információ *be- és kivitelét*.

A tervező rendszer bármely számítógépre történő egyszerű átvitelének biztosítása érdekében az adatkezelő rendszerrel szemben is támasztott természetes követelmény volt az alkalmazott számítógép speciális tulajdonságainak minimális mértékű kihasználása.

Mivel az ismert listakezelő rendszerek ([1], [2], [3]) a fenti követelményeknek csak egy részét elégíti ki, a megfogalmazott feladatok megoldására készül a LIDI (LIst processing system of the department of DIgital techniques), amelynek 1972 évi változatában kidolgozott:

- az attributumokkal ellátott szögpontokkal rendelkező általános szerkezetű irányított gráfok belső reprezentációja;
- a gráfokon definiált legfontosabb általános műveleteket realizáló programok gyűjteménye;
- az attributumokkal ellátott szögpontokkal rendelkező erdők leírására szolgáló nyelv;
- a bemenő nyelvet értelmező, a leírt erdőt bekapcsoló program, valamint az információ kiadására szolgáló primitív program.

Az információ belső ábrázolása az attributumokkal rendelkező általános szerkezetű irányított gráfok *közvetlen* reprezentációját lehetővé tevő olyan kapcsolt lista, amelynél az egyes szögpontokhoz tartozó attributumok a megfelelő elem mellett sorosan helyezkednek el.

A gráfokon definiált általános műveleteket realizáló, valamint az információ be- és kivitelét biztosító programok USASI FORTRAN nyelven írt szubrutinok, amelyek az MTA CDC 3300 számítógépén használt MASTER operációs rendszer segédkönyvtárában állnak a felhasználók rendelkezésére.

A rendszer bemenő nyelve lehetővé teszi az erdők leírását különböző formátumokban, amely formátumok egymásba is skatulyázhatók. A nyelv lehetőséget ad a már belül reprezentált gráf azon szögpontjának a szögponthoz vezető valamely út leírásával történő kijelölésére is, amely alá a leírt erdőt be kell kapcsolni.

A LIDI-72 rendszer kidolgozásában egyes programok elkészítésével résztvett János József, az ELTE matematika szakos hallgatója, valamint bizonyos technikai munkák elvégzésével Sebestyén Ilona, a Digitális Technika Osztály dolgozója.

A rendszer e rövid ismertetése tartalmazza a belső reprezentáció, a kidolgozott programok és a bemenő nyelv leírását, valamint a rendszer fejlesztésével kapcsolatos feladatok felsorolását.

2. A BELSŐ REPRESENTÁCIÓ

A választott belső reprezentációban a lista egy eleme által elfoglalt szavak száma függ az elem típusától és elhelyezésétől, de a lista minden eleme rendelkezik egy egyszavas *fej*vel. Bármely elemre való hivatkozás az elem fejére történő hivatkozást jelent; egy szóra pedig az indexével (címével) hivatkozunk.

Egy szóban az elemnek általában több mezője helyezkedik el, amelyek közül a választott belső reprezentáció szempontjából fontos szerepet játszanak a *jelzések*hez rendelt mezők. A jelzések logikai függvények, amelyeknek értelmezési tartománya szerint két típusát különböztetjük meg. A jelzések egyik típusát jelentik azon $J1(x)$ függvények, amelyek értelmezési tartománya a lista elemei halmazának valamely részhalmaza. Az ilyen $J1$ jelzésnek az x elemnél felvett pillanatnyi értékét az x elem fejében szereplő, a $J1$ jelzéshez hozzárendelt mező tartalmazza. Az egyszerűség érdekében, ha a lista egy x eleme hozzátartozik a $J1$ értelmezési tartományhoz és $J1(x) = 1$, akkor azt mondjuk, hogy az x elem $J1$ jelzéssel szerepel, és az elem ábrázolásánál a $J1$ jelzést az elem fejében feltüntetjük. A jelzések másik típusát képezik azon $J2(s)$ jelzések, amelyek értelmezési tartománya a lista elemei által elfoglalt szavak halmazának valamely részhalmaza. Az ilyen $J2(s)$ jelzésnek az s szónál felvett értékét az s szóban szereplő, a $J2$ jelzéshez hozzárendelt mező tartalmazza. A továbbiakban, ha egy s szó hozzátartozik a $J2$ jelzés értelmezési tartományához és $J2(s) = 1$, akkor azt mondjuk, hogy az s szóban a $J2$ jelzés szerepel és a szó ábrázolásánál a $J2$ jelzést feltüntetjük.

2.1 AZ INFORMÁCIÓ ÁBRÁZOLÁSA

A LIDI-72 rendszernél az információ belső reprezentációjában az irányított gráf minden szögpontjának megfelel a lista egy és csakis egy *szögpontelem*e. Minden szögpontelemre értelmezett a következő négy *jelzés*:

$S(x)$, amely akkor és csak akkor 0, ha az x szögpontelem;

$T(x)$ – annak logikai értéke, hogy az x elemnek megfelelő X szögpont terminális;

$A(x)$ – annak logikai értéke, hogy az x elemnek megfelelő X szögponthoz van rendelve attributum;

$E(x)$ – a gráfon végzett műveleteket realizáló programokban használt vizsgálati jelzés.

Minden szögpontelem fejében – a fenti négy jelzéshez rendelt jelzésmezőn kívül – az elem névmezője helyezkedik el, amely az elem *nevét* tartalmazza. Egy szögpontelem neve a névmezőben elhelyezkedő olyan nemnegatív egész szám, amely vagy az input program által rendelődött a bemenő információban leírt valamely szögpontnévhez, vagy eleme a rendszer számára fenntartott elemnevek halmazának.

Ha egy szögpontelem A jelzéssel szerepel, akkor az elem fejét közvetlenül megelőző szóban fel van tüntetve az elemnek megfelelő szögponthoz rendelt *attributumok* N száma; az azt megelőző N szóban pedig az attributumok értékei helyezkednek el,

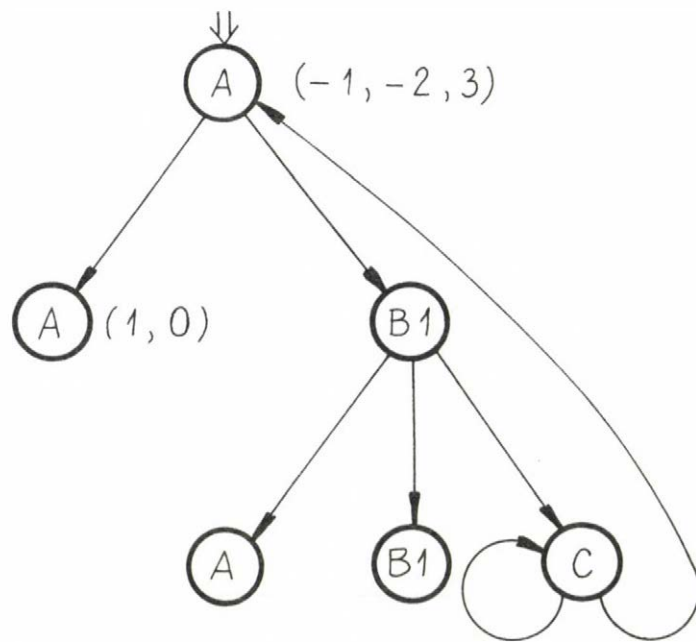
A belső reprezentációban az irányított gráf valamely nem terminális szögpontjából kiinduló minden élnek megfelel a szögpontnak megfelelő (T jelzés nélkül szereplő) szögpontelem egy és csak is egy *élpontere*, amely az él végpontját jelentő szögpontnak megfelelő szögpontelemre hivatkozik (mutat), azaz a reprezentáció egyirányú lista. Egy nem terminális szögpontelem élponterei egy lineáris listát alkotnak, amelynek minden elemét egy-egy szó pointermezője tartalmazza.

E pointerlista a szögpontelem fejét követő szótól kezdve helyezkedik el; az utolsó élpontert tartalmazó szó L jelzéssel szerepel. Egy szögpontelem élpontereinek listája legegyszerűbb esetben soros elhelyezésű. Az információ struktúra módosításának egyszerűsítése érdekében azonban lehetséges egy szögpontelem pointerlistájának szakaszonként soros elhelyezése; ekkor a pointerlista sorosan elhelyezett pointerekből álló pointerszakaszok kapcsoltan elhelyezett lineáris listája. Egy $k > 0$ számú egymásutáni szóban elhelyezkedő nem utolsó pointerszakasz első $k - 1$ szavában élponterek szerepelnek, C jelzéssel ellátott utolsó szavának pointermezője pedig a következő pointerszakaszra mutató *folytatáspointert* tartalmazza. (Az utolsó pointerszakasz végét az utolsó élponternél szereplő L jelzés jelzi.) A fentiek szerint a szögpontelemek pointerreit tartalmazó szavakon a következő jelzések értelmezettek:

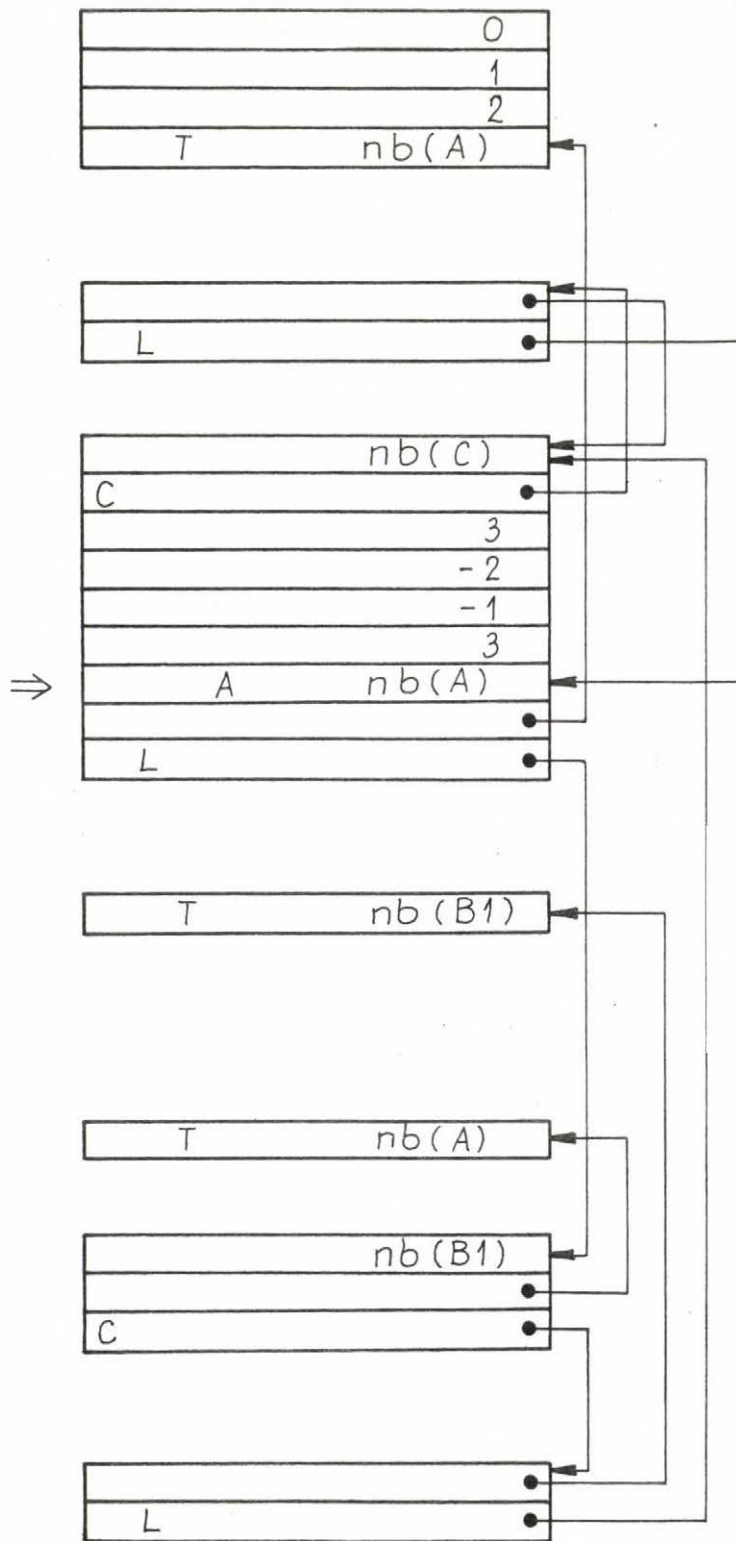
$C(p)$ – annak logikai értéke, hogy a p szó pointermezőjében folytatáspointer szerepel;

$L(p)$ – annak logikai értéke, hogy a p szó pointermezőjének tartalma szögpontelem utolsó élpontere.

A fentieknek megfelelően az 1. ábrán szereplő irányított gráf belső reprezentációja a 2. ábrán szemléltetett lista lehet, ahol $nb(A)$, $nb(B1)$, illetve $nb(C)$ jelöli az A , $B1$, illetve C szögpontnevekhez rendelt belső elemneveket.



1. ábra



2. ábra

2.2 LÉPTETŐELEMÉK

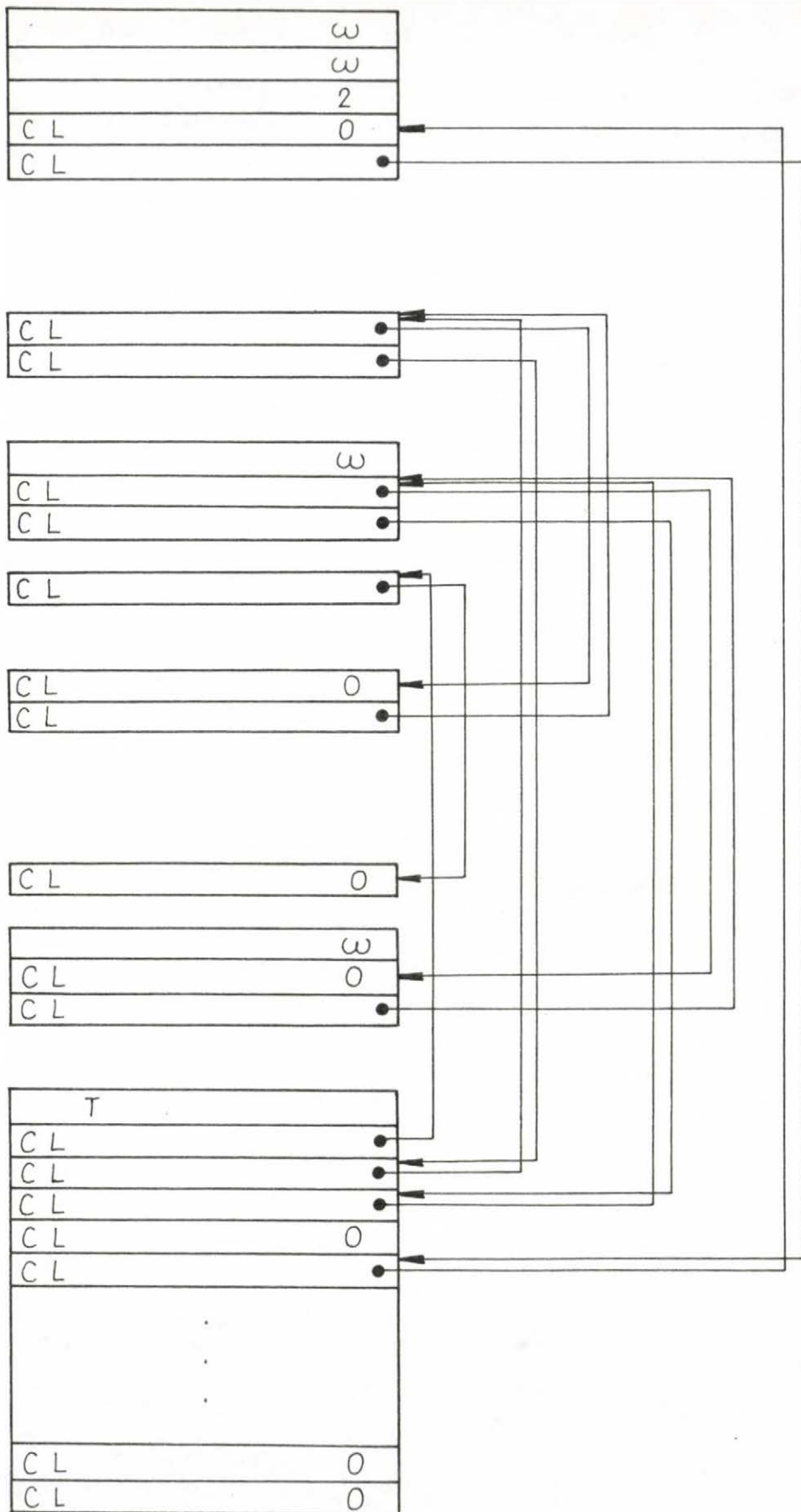
A szögpontelemek pointereinek és attributumainak az elem fejtől kezdődő, valamint az attributumok soros elhelyezése miatt a lista által reprezentált irányított gráfon végzett olyan operációk, amelyek során terminális szögpontról éleket indítunk ki, vagy valamely szögponthoz rendelt attributumok halmazát bővítjük, szükségessé tehetik az érintett szögpontra megfelelő szögpontelem áthelyezését nagyobb összefüggő szabad területre. Annak érdekében, hogy ilyen esetekben elkerülhető legyen az áthelyezett szögpontelemre mutató minden pointernek (az egyirányú reprezentáció miatt) a lista teljes bejárásával történő megkeresése és módosítása, az áthelyezett szögpontelem eredeti helyén egy csak fejből álló *léptetőelem* marad, amely az áthelyezett szögpontelemre mutató, *S* jelzéssel ellátott pointer.

2.3 A SZABAD MEZŐK NYILVÁNTARTÁSA

Nyilvánvaló, hogy a választott belső reprezentációban a lista rendelkezésére álló összefüggő területben a reprezentált irányított gráf szögpontjainak megfelelő szögpontelemek és az esetleg előforduló léptetőelemek által el nem foglalt (a gráf struktúrájának módosításával is előálló) szabad szavak különböző hosszúságú összefüggő *szabad mezőket* alkotnak. A lista rendelkezésére álló összefüggő terület gazdaságos felhasználása érdekében a szabad mezők kezelése figyelembe veszi e mezők hosszát. A szabad mezők kezelésének alapja a szabad mezők *katalógusa*, amely a listának egy terminális jelzéssel ellátott kitüntetett eleme és amely az $1, 2, \dots, 20$, valamint a 20-nál hosszabb szabad mezők listáinak első elemeire mutató, *CL* jelzéssel ellátott pointeret tartalmazza. Minden szabad mező a szabad mezők katalógusából induló, az adott hosszúságú szabad mezőket tartalmazó – természetesen kapcsolt elhelyezésű – *lineáris listának* eleme. E listák minden eleme (azaz minden szabad mező) rendelkezik egy *CL* jelzéssel ellátott fejjel, amelyben – az utolsó elem kivételével – az adott lista következő elemére mutató pointer szerepel. Minden lista utolsó elemének feje nullpointert tartalmaz; hasonló módon a szabad mezők valamely listájának üres voltát a katalógusban nullpointer jelzi. Az egy hosszúságú szabad mezők listájában minden elem csak fejjel rendelkezik, tehát e lista egyirányúan kapcsolt lista. Az egy-nél hosszabb szabad mezők listáiban minden elem tartalmaz egy *CL* jelzéssel ellátott pointeret, amely első elem esetén a szabad mezők katalógusának megfelelő szavára, a többi elemnél pedig az elemet az adott listában megelőző elemre mutat, így az egy-nél hosszabb szabad mezők listái kettősen kapcsolt listák. A három hosszúságú szabad mezőknél a fejet megelőző szóban egy – attributumként kis valószínűséggel szereplő – ω érték szerepel. A $k > 3$ hosszúságú szabad mezők esetén – a szögpontelemek attributumaihoz hasonlóan – a fej előtti szóban a $k - 3$ érték, az azt megelőző $k - 3$ szó mindegyikében pedig ω helyezkedik el. Látható, hogy a szabad mezőkben tárolt adatok lehetővé teszik a szabad mezőkhöz tartozó szavak felismerését a lista számára rendelkezésre álló terület lineáris vizsgálatával a magasabb indexek felől az alacsonyabb indexek felé haladva biztosan, ellenkező irányban pedig nagy valószínűséggel, továbbá

biztosítják a szabad mezők listáinak veremként való kezelésén kívül az egynél hosszabb szabad mezők listáiban a közbülső elemek kiiktatásának lehetőségét is.

A 3. ábra a szabad mezők listáinak szerkezetére mutat példát.



totalógus ⇒

3. ábra

3. MŰVELETEK A LISTÁN

Az előzőkben ismertetett listán végezhető műveleteket FORTRAN nyelven írt, az MTA CDC 3300 számítógépének segédkönyvtárában tárolt szubrutinok rendszere valósítja meg.

3.1 KEZDETI ÁLLAPOT BEÁLLÍTÁSA

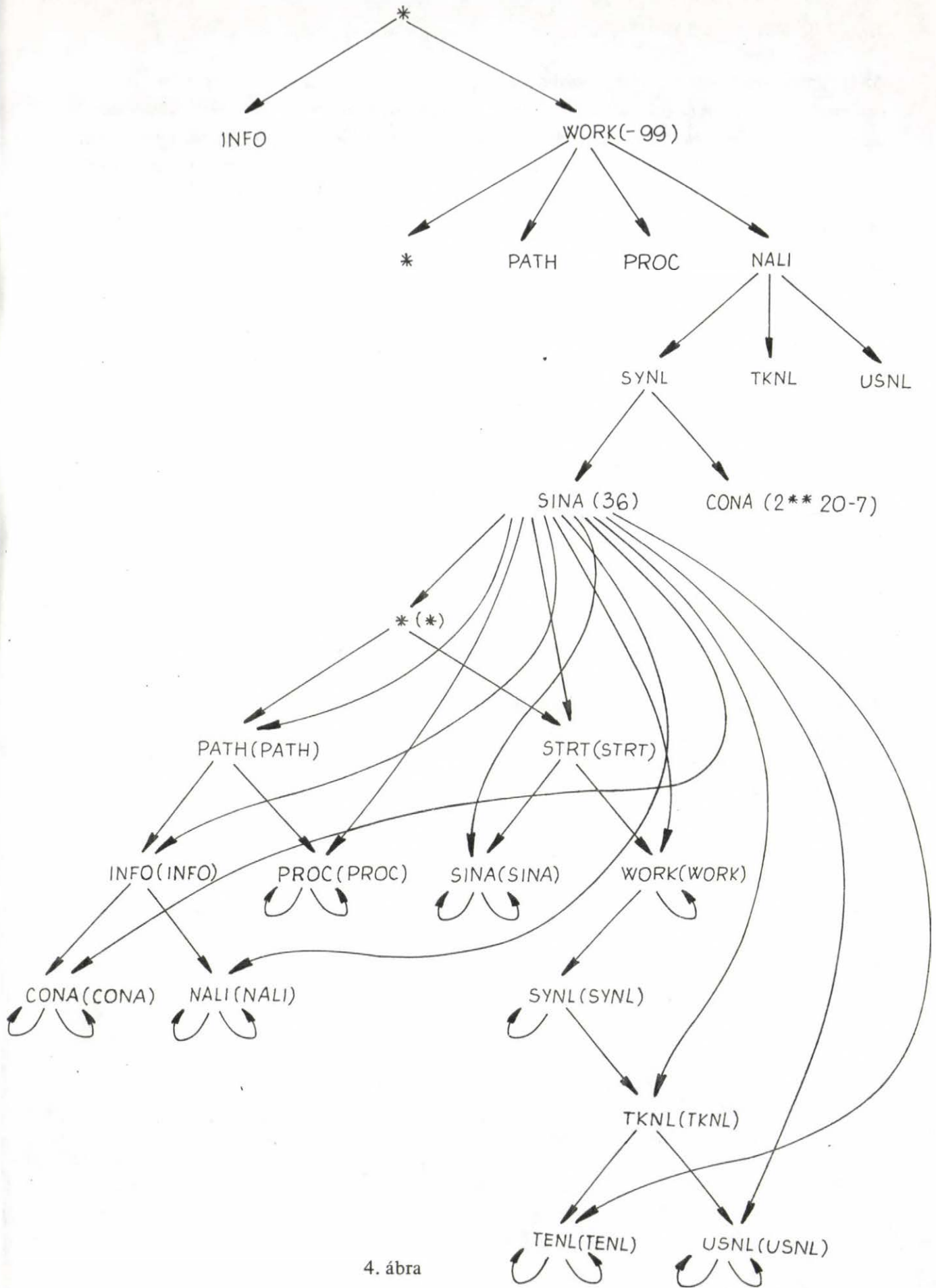
A kezdeti állapot beállítása a *BLDA* és a *GEDA* nevű szubrutinok e sorrendben való hívásával történik. E két rutin beállítja a számítógéptől (pl. szóhosszától, az operatív tároló méretétől) függő változók értékeit, valamint egyes COMMON területekben használt változók alapértékeit. Ezenkívül a *GEDA* szubrutinnal beállítható a lista alapállapota, amely jelenleg a 4. ábrán szemléltetett irányított gráf belső reprezentációja, ahol a * megnevezetlen szögpontot jelöl. A

*, *WORK*, *NALI*, *SYNL*, *SINA*

szögpontnevek sorozatával meghatározott út végpontja alatti gráf a rendszer indításához szükséges egyszerű rendszernevek gráfja, amelynek szögpontjainál attributumként e nevek szerepelnek. A

*, *WORK*, *

út végpontja a szabad mezők katalógusa, amely a lista alapállapotában egyetlen szabad mezőként a szemléltetett gráf reprezentációja mellett megmaradó összefüggő szabad területet tartalmazza.



4. ábra

3.2 AZ INFORMÁCIÓ BEVITELE

Az információ bevitele az *INPT* szubrutin segítségével történik, amelynek egyszeri hívásával beolvásodik az *1* nevű (általában az *INP*) file-ról és értelmezésre kerül a LIDI-72 rendszer bemenő nyelvének egy *előírása*. (A rendszer bemenő nyelvének leírását a 4. fejezet tartalmazza.) Az előírásban szerepelhet beviendő információ leírása a bemenő nyelvben megengedett ötféle formátumban. Egy információleírás valamely irányított erdő leírása, amelynek belső reprezentációját az *INPT* szubrutin bekapcsolja a listának azon szögpontelem alá, amely az adott előírásban meghatározott út végpontjának felel meg. Egy előírt út végpontjának megfelelő szögpontelem megkeresését az *INPT* szubrutin a – közvetlen hívással is alkalmazható – *PASE* szubrutin segítségével végzi. Működése során az *INPT* szubrutin előállítja a külső szögpontnevekkel és – az egyes formátumok szerint használt – különböző elhatárolókkal leírt erdő belső elemneveket tartalmazó zárójeles leírását. Az előírás értelmezése során a belső neveket az egyszerű nevekhez a *SINI*, az összetett nevekhez pedig a *CONI* függvény rendeli; a külső nevekhez már előzőleg rendelt belső nevek megkeresésére a – közvetlenül is használható – *SINS*, illetve *CONS* függvények szolgálnak. A beviendő erdő zárójeles leírásának befejezésére és az információ bekapcsolására az *INPT* szubrutin hívja az *IFIN* szubrutint, amely az információnak a listába történő bekapcsolását a megfelelő bekapcsoló szubrutin segítségével végzi.

3.3 INFORMÁCIÓ BEKAPCSOLÁSA

A LIDI-72 rendszerben valamely *erdő* belső reprezentációjának az erdő egy COMMON területben előállított zárójeles leírása alapján a listába történő bekapcsolására több szubrutin (*TRIN*, *FRIN*, *FRNA*) is szolgál, amelyek lényegében az általuk bekapcsolható erdő méreteiben térnek el egymástól. E szubrutinok használják az egy *szögpontelem* bekapcsolását végző *NOIN*, illetve *NINA*, valamint az egy szögpontból kiinduló *éleknek* megfelelő élpointerek bekapcsolására szolgáló *BRIN*, illetve *BRNA* szubrutinokat. Ez utóbbi szubrutinok közvetlen hívásával valósítható meg valamely – pl. az *INPT* szubrutinnal bevitt – erdő belső reprezentációjának olyan módosítása is, amelynek eredményeként előálló lista egy bonyolultabb (pl. ciklusokat is tartalmazó) gráf belső reprezentációja lesz. Egy szögponthoz rendelt *attributumok* halmazának bővítése a belső reprezentációban – egymástól a bővítő attributumok megadásának módjában különböző – *ATIN*, *AINN* és az *AINL* szubrutinok felhasználásával történik. Egy COMMON területben lévő – pl. egy háttértárolóból beolvasott – listának, vagy valamely részének a listába való bekapcsolása a *LIIN* szubrutin segítségével valósítható meg.

3.4 INFORMÁCIÓ TÖRLÉSE

A rendszerben valamely *gráf* belső reprezentációjának a listából való törlése a *GRRE* vagy a *GRER* szubrutin felhasználásával történhet; a *GRRE* szubrutin egy adott szögpontról elérhető részgráf belső reprezentációját a kiindulási szögpontról megtartásával, a *GRER* szubrutin e szögpontról együtt törli. E szubrutinok felhasználják az egy szögpontra megfelelő *szögpontelem* törlését végző *NORE* és az egy szögpontról kiinduló *éleknek* megfelelő élpointerek törlését megvalósító *BRRE* szubrutinokat. Az élpointerek törlése az *MBRE*, *BRRW*, *PTSR*, *MBSB* vagy *MBSW* szubrutinok segítségével is történhet, amelyek a törlendő élek meghatározásában, illetve a szubrutinokban felhasznált munkaterületek méreteiben különböznek. Valamely szögponthoz rendelt *attributumok* törlésére az *ATRE*, *ATCR* és *ATRW* szubrutinok szolgálnak, amelyek a törlendő attributumok megadásának módjában, valamint a felhasznált munkaterület méreteiben térnek el egymástól.

3.5 SZABAD MEZŐK KEZELÉSE

Az információnak mind a bekapcsolását, mind a törlését végző szubrutinok használják a szabad mezőket kezelő szubrutinokat; az előbbieket – természetesen – elsősorban a szabad mezők lefoglalását, az utóbbiak pedig a mezők felszabadítását végző szubrutinokat hívják. A választott belső reprezentációban azonban a szabad mezőket kezelő szubrutinok e két típusa nem alkot teljesen különálló csoportot: valamely szabad mező egy részének lefoglalása esetén a fennmaradó szavakból álló mező bekapcsolódik a szabad mezők megfelelő (rövidebb mezőket tartalmazó) listájába; egy mező felszabadítása során pedig a mezővel esetleg szomszédos szabad mezők kiiktatódnak a szabad mezők listáiból és az így kialakított (hosszabb) mező kapcsolódik be a szabad mezők megfelelő listájába.

A szabad mezők *lefoglalását* végző legmagasabb szintű szubrutin az *FPMM*, amelynek felhasználásával lefoglalható egy szögpontelem elhelyezéséhez szükséges hely, ha léteznek legalább a szakaszonként soros elhelyezést biztosító szabad mezők. A szükséges szabad mezők keresése két paraméter alapján történik: a *MAX* paraméter megadja az egyetlen összefüggő mezőbe történő elhelyezés esetén szükséges szavak számát, a *MIN* paraméter értéke pedig az egybefüggő területre elhelyezendő szavak minimális száma; a *MAX-MIN* számú szó lefoglalását az *FPMM* szubrutin legalább kettő hosszúságú mezők lefoglalásával biztosítja a megfelelő folytatáspointerek elhelyezése mellett. Az *FPMM* szubrutin természetesen használja az adott hosszúságú szabad mező keresését és a szabad mezők megfelelő listájából való kiiktatását végző *FPSO*, valamint (a fennmaradó részmező bekapcsolására) a mezők bekapcsolását megvalósító *FPIN* szubrutint. Az *FPSO* szubrutin egy konkrét szabad mezőnek a megfelelő listából történő kiiktatását az *FPOU* szubrutin felhasználásával végzi.

Valamely, a legalacsonyabb és legmagasabb indexével adott mező *felszabadítása* a *PORE* szub-

rutin hívásával történik, amely az *FPFO* és az *FPBO* szubrutinokkal megkeresteti (és a megfelelő listából kiiktattatja) a mezőhöz csatlakozó szabad mezőket, majd az így kialakított mezőt az *FPIN* szubrutin segítségével bekapcsolja a szabad mezők megfelelő listájába.

Nyilvánvaló, hogy a választott belső reprezentációban a szabad mezők előbb ismertetett kezelése mellett is előfordulhat a szabad helyek olyan mértékű elaprózódása, hogy egy – pl. több attribútummal rendelkező – szögpontelem elhelyezése nem oldható meg, bár a lista számára rendelkezésre álló területben van elegendő szabad hely. Nyilvánvaló továbbá, hogy az élpointerek szakaszonként soros elhelyezésénél használt folytatáspointerek, valamint a szögpontelemek elléptetése során keletkezett léptetőelemek a lista által reprezentált irányított gráf szempontjából feleslegesen foglalják a helyet a lista számára fenntartott területben. Feleslegesen foglalja a helyet a lista számára rendelkezésre álló területben az olyan allista is, amelyet a felhasználó nem törölt, bár az allistára történő minden hivatkozás megszűnt; a választott belső reprezentációban ugyanis – mivel a szögpontelemek hivatkozásszámlálót nem tartalmaznak – automatikus törlés nem lenne gazdaságos. A felesleges helyfoglalások kiküszöbölése és a maximális összefüggő szabad terület kialakítása az információ *tömörítése* útján történik. A tömörítő szubrutinok a tömörítendő lista minden (elérhető) szögpontelemét – a nem terminális elemek pointerlistáinak soros elhelyezése mellett – valamely összefüggő terület egymás utáni szavaira helyezik el. Mivel a választott belső reprezentációban egy szögpontelem által elfoglalt szavak száma változó, a lista tömörítése nem oldható meg a jól ismert "garbage collection" algoritmussal [4]; a LIDI-72 rendszer tömörítő szubrutinjai egyszerű, de nagy munkaterületet igénylő algoritmusokkal működnek. Az allisták tömörítésére szolgáló *COMP* szubrutin munkaterülete operatív tárolóbeli terület, a *COMF* szubrutin pedig file-t használ munkaterületként. E szubrutinok lehetővé teszik az adott területben mind az alacsonyabb indextől a magasabbak felé haladó, mind az ellenkező irányú tömörítést, valamint kijelölt szögpontelemek feje tömörítés utáni indexének meghatározását. A tömörítés befejezése után mindkét szubrutin az *FPCI* szubrutin segítségével alapállapotba hozza a szabad mezők katalógusát, amelyben a lista számára rendelkezésre álló területnek a tömörítés után fennmaradt része szerepel egyetlen összefüggő szabad mezőként. A *COMP* illetve a *COMF* szubrutin felhasználásával a *TOCO* illetve a *TOCF* szubrutin végzi a teljes lista meghatározott szabályok szerinti tömörítését.

3.6 BEJÁRÁSOK

A LIDI-72 rendszer szubrutinjainak külön csoportját alkotják a lista által reprezentált irányított gráf valamely részgráfjának bejárásait realizáló szubrutinok. E szubrutinok a bejárás során az adott részgráf szögpontjainak megfelelő szögpontelemek mindegyikénél egyszer és csak egyszer elvégeztetik a felhasználó által előírt, külső szubrutinnal adott operációt, amely a *TOOP*, a *LEOP* és a *LTOP* szubrutinok esetén *feltétel nélküli*, a többi bejáró szubrutinnál pedig *feltételes* lehet. A *TOOP* és a *CTOA* szubrutinok bejárják a lista által reprezentált irányított gráf

egy adott szögpontjából elérhető szögpontok által meghatározott részgráfnak megfelelő *teljes* listát, a *CTOP* szubrutin e lista bejárását megszakítja, amint a felhasználó által adott külső szubrutin paraméterként szereplő Boole-változó értéke "hamis" lesz. A *LEOP* és a *CLTA* szubrutinok a gráf egy adott szögpontjából meghatározott *szinten* lévő összes szögpont bejárását realizálják, a *CLOP* szubrutin viszont – a *CTOP* szubrutinhoz hasonlóan – abbahagyja a bejárást, amint a Boole-változó "hamis" értéket vesz fel. Az előző két csoportbeli szubrutinhoz hasonló viszonyban vannak egymással a *LTOP* és a *CLTA*, illetve a *CLTO* szubrutinok, amelyek a lista által reprezentált irányított gráf kitüntetett szögpontjából adott *szintig* bezárólag elérhető szögpontok által meghatározott részgráf bejárásait realizálják.

3.7 SZERKEZETI VIZSGÁLATOK

A LIDI-72 rendszer szubrutinjainak önálló csoportját képezik a lista által reprezentált irányított gráf szerkezetét elemző szubrutinok, amelyeket János József diplomamunkája keretében dolgozott ki. E szubrutinok segítségével eldönthető, hogy a gráf egy kitüntetett szögpontjából elérhető szögpontok által meghatározott részgráf *erdő*-e (*TRCH*), vagy legalább bármely szögpontjának *szintje* egyértelmű-e (*LECH*), illetve *ciklusmentes*-e (*CYCH*). E csoporthoz tartozó *DRUG* szubrutin a lista által reprezentált irányított gráf egy adott szögpontjából elérhető szögpontok által meghatározott részgráf *indexcsomópontrendszerének* meghatározására szolgál.

4. A BEMENŐ NYELV LEIRÁSA

4.1 ALAPJELEK, ELHATÁROLÓK, ELEMÉK

A LIDI-72 bemenő nyelvének mondatai a latin abc nagybetűiből és a tizes számrendszer számjegyeiből elválasztók segítségével épülnek fel. Elválasztóként szerepelnek a $.$, $/$, $:$, $()$ írásjelek, az előjelek és az egyenlőségjel, a $*$ és a $\%$, bizonyos esetekben az *üresjel* (szóköz), valamint a lyukkártyák használatával kapcsolatos *1. oszlop* alapjel, amely azt jelenti, hogy a 6. oszlop előtt *üresjeltől* különböző – feldolgozandó – karakter fordul elő. A nyelv alapjelkészlete tehát a következő:

⟨alapjel⟩ ::= *üresjel* | ⟨nem üresjel⟩

⟨nem üresjel⟩ ::= ⟨betű⟩ | ⟨számjegy⟩ | ⟨elválasztó⟩

⟨betű⟩ ::= A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z

⟨számjegy⟩ ::= 0|1|2|3|4|5|6|7|8|9

⟨elválasztó⟩ ::= *1.oszlop* | $.$ | $/$ | $:$ | $($ | $)$ | $=$ | $*$ | $\%$

A bemenő nyelv minden mondata külön sorba (*1.oszloppal* kezdve) írandó, és egy sorban (az *1.oszlop*ot nem számítva) maximálisan 72 alapjel szerepelhet. Ha valamely mondat 72-nél több alapjeltől áll, akkor több sorba írható: a mondat első sorát tetszésszerű számú folytatássor követheti. Folytatássorba elválasztóknál térhetünk át. A folytatássort 5 üresjelet követően a 6. oszlopba írt – feldolgozásra nem kerülő – nem üresjeltől álló folytatásjel jelzi:

⟨folytatásjel⟩ ::= ⟨5 üresjel⟩⟨nem üresjel⟩

⟨5 üresjel⟩ ::= ⟨4 üresjel⟩*üresjel*

⟨4 üresjel⟩ ::= ⟨2 üresjel⟩⟨2 üresjel⟩

⟨2 üresjel⟩ ::= *üresjel üresjel*

Az áttekinthetőség fokozása érdekében bármely elválasztó előtt vagy után tetszésszerű számú *üresjel* alkalmazható; a nyelv mondatainak felépítésénél az így, valamint a folytatásjelek felhasználásával kialakított összetett elhatárolók játszanak szerepet. Elhatárolóként szerepelnek még a közvetlenül egymás után írt (= és az =), mint összetett kezdő- illetve végzárójel, valamint a $:$ -tal azonos jelentésű két pont, továbbá a lyukkártyák használatával kapcsolatos 7. oszlop, amely azt jelenti, hogy az első *üresjeltől* különböző karakter a 6. oszlop után szerepel. Összetett kezdő- és végzárójelen belül pedig tetszésszerű, számjegyeket és előjeleket, valamint összetett kezdő- és végzárójelet nem tartalmazó jelsorozatok használhatók elhatárolóként.

⟨elhatároló⟩ ::= ⟨üresjel⟩ | ⟨pont⟩ | ⟨vessző⟩ | ⟨törtvonal⟩ | ⟨kettőspont⟩

⟨nyitózárójel⟩ | ⟨végzárójel⟩ | ⟨pluszjel⟩ | ⟨minuszjel⟩ |

⟨egyenlőségjel⟩ | ⟨csillag⟩ | ⟨százalékjel⟩ |

⟨összetett kezdőzárójel⟩ | ⟨összetett végzárójel⟩ |

⟨7.oszlop⟩

⟨üresjel⟩ ::= *üresjel* | ⟨üresjel⟩*üresjel* | ⟨üresjel⟩ ⟨folytatásjel⟩

<pont> ::= . | <pont>üresjel | <pont> <folytatásjel>
 <vessző> ::= , | <vessző>üresjel | <vessző> <folytatásjel>
 <törtvonal> ::= / | <törtvonal>üresjel | <törtvonal> <folytatásjel>
 <kettőspont> ::= : | . | <kettőspont>üresjel | <kettőspont> <folytatásjel>
 <pluszjel> ::= + | <pluszjel>üresjel | <pluszjel> <folytatásjel>
 <mínuszjel> ::= - | <mínuszjel>üresjel | <mínuszjel> <folytatásjel>
 <egyenlőségjel> ::= = | <egyenlőségjel>üresjel | <egyenlőségjel> <folytatásjel>
 <kezdőzárójel> ::= (| <kezdőzárójel>üresjel | <kezdőzárójel> <folytatásjel>
 <végzárójel> ::=) | <végzárójel>üresjel | <végzárójel> <folytatásjel>
 <csillag> ::= * | <csillag>üresjel | <csillag> <folytatásjel>
 <százalékjel> ::= % | <százalékjel>üresjel | <százalékjel> <folytatásjel>
 <összetett kezdőzárójel> ::= (= | <összetett kezdőzárójel> <számelhatároló>
 <összetett végzárójel> ::=) | <számelhatároló> <összetett végzárójel>
 <számelhatároló> ::= üresjel | <betű> | 1.oszlop | . | , | / | : | (|) | * | % | <7.oszlop> | <folytatásjel> |
 <számelhatároló> <számelhatároló>

A nyelv alapelemeiből elhatárolók segítségével épülnek fel a nyelv összetettebb elemei – végső soron a mondatok. A nyelv alapelemei az elemi nevek és a természetes számok. Az elemi nevek legfeljebb 8 alfanumerikus alapjélekből álló jelsorozatok. A természetes számok számjegyek olyan sorozatai, amelyek numerikus értékei nem haladják meg az adott számítógép választott hosszúságú szavában tárolható pozitív egész számok maximális értékét – ez a CDC 3300 gépre készült reprezentációban $2^{*}23-1$. A korlátozások figyelmen kívül hagyásával:

<elemi név> ::= <betű> | <számjegy> | <elemi név> <betű> | <elemi név> <számjegy>
 <természetes szám> ::= <számjegy> | <természetes szám> <számjegy>

A hatástalan karakterek megengedését a következő szabály fejezi ki:

<egyszerű név> ::= <elemi név> | <egyszerű név>üresjel | <egyszerű név> <folytatásjel>

A nyelv lehetővé teszi az egyszerű neveken kívül ezekből összetett nevek használatát is. A jelenlegi reprezentációban egy összetett nevet alkotó komponens egyszerű nevek száma maximálisan 197 (általában SE-3) lehet.

A nyelvben lehetőség van előjeles egész számok használatára is – a CDC 3300 gépre készült reprezentációban a szám abszolút értéke legfeljebb $2^{*}23-1$ lehet. A korlátozások figyelmen kívül hagyásával a nyelv mondatainak alkotóelemei:

<elem> ::= <név> | <szám>
 <név> ::= <egyszerű név> | <összetett név>
 <összetett név> ::= <egyszerű név> <pont> <egyszerű név> | <összetett név> <pont> <egyszerű név>
 <szám> ::= <természetes szám> | <pluszjel> <természetes szám> | <mínuszjel> <természetes szám>

⟨számsorozat⟩ ::= | ⟨szám⟩ | ⟨számsorozat⟩⟨számelhatároló⟩⟨szám⟩

⟨számhalmaz⟩ ::= ⟨összetett kezdőzárójel⟩

⟨számsorozat⟩

⟨összetett végzárójel⟩

4.2 MONDATOK, ELŐÍRÁS

A LIDI-72 bemenő nyelvének mondatait alkotják az információ leírások, az útelőírások, az eljáráselőírások és a megjegyzések. Az információ leírások a beviendő információt adják meg; az útelőírás meghatározza azt a szögpontot, amely alá a következőkben leírt információt be kell kapcsolni; az eljáráselőírások elvégzendő műveleteket jelölnék ki, a megjegyzések pedig hatástalanok. A nyelv mondatainak előírás vége jellel lezárt sorozata egy előírást képez.

⟨előírás⟩ ::= ⟨előírás vége jel⟩ | ⟨mondat⟩⟨előírás⟩

⟨előírás vége jel⟩ ::= *l.oszlop*⟨pluszjel⟩

⟨mondat⟩ ::= ⟨információ leírás⟩ |

⟨útelőírás⟩ |

⟨eljáráselőírás⟩ |

⟨megjegyzés⟩

4.2.1 AZ INFORMÁCIÓ LEIRÁSA

A LIDI-72 rendszerben a leírt információ értelmezése kétféle, lényegesen különböző módon történik. Az alapvető értelmezés szerint az információ leírásában szereplő elhatárolók az elhelyezendő információ szerkezetét kifejező jelekként kerülnek értelmezésre; az ilyen értelmezést kívánó információ az alapformátumok valamelyikén írható le. A leírt információ másik értelmezése szerint – tetszésszerű értelmezés lehetőségének biztosítása céljából – az elhatárolók is elhelyezésre kerülnek; az ilyen értelmezést igénylő információ kiegészítő formátumban írható le.

⟨információ leírás⟩ ::= ⟨alapformátumú információ leírás⟩ |

⟨kiegészítő formátumú információ leírás⟩ |

⟨információ leírás⟩⟨alapformátumú információ leírás⟩ |

⟨információ leírás⟩⟨kiegészítő formátumú információ leírás⟩

4.2.1.1 ALAPFORMÁTUMOK; SZÖGPONTOK LEIRÁSA

A LIDI-72 bemenő nyelvében a beviendő információ leírására szolgáló alapformátumok lehetővé teszik olyan erdők leírását, amelyek bármely szögpontjához tetszésszerű számú, egész értékű attributum lehet rendelve. Az ilyen információ leírására négyféle alapformátum áll rendelkezésre.

zésre: a zárójeles és a léptetőjeles szabad formátum, valamint a négy, illetve a hat szint leírására szolgáló kötött formátum. Lehetőség van az alapformátumok egymásba skatulyázására. Az alapformátumokban meghatározott elhatárolókkal elválasztva egyes szögpontok leírását kell megadni. Egy szögpont teljes leírása a szögpont nevének és a szögpontokhoz rendelt attributumhalmaznak a megadásából áll. Az üres attributumhalmaz elhagyható. Elhagyható – a léptetőjeles formátumú leírás kivételével – a szögpont neve is, ha az az információ kezelésében nem játszik szerepet.

⟨alapformátumú információ leírás⟩ ::= *I.oszlop* ⟨szabad formátumú erdőleírás⟩ |
⟨kötött formátumú erdőleírás⟩ |
⟨alapformátumú információ leírás⟩
I.oszlop ⟨szabad formátumú erdőleírás⟩ |
⟨alapformátumú információ leírás⟩
⟨kötött formátumú erdőleírás⟩

⟨szabad formátumú erdőleírás⟩ ::= ⟨zárójeles erdőleírás⟩ | ⟨léptetőjeles erdőleírás⟩ |
⟨szabad formátumú erdőleírás⟩⟨zárójeles erdőleírás⟩ |
⟨szabad formátumú erdőleírás⟩⟨léptetőjeles erdőleírás⟩

⟨kötött formátumú erdőleírás⟩ ::= ⟨fa 4 szintjének kötött formátumú leírása⟩ |
⟨fa 6 szintjének kötött formátumú leírása⟩ |
⟨kötött formátumú erdőleírás⟩
⟨fa 4 szintjének kötött formátumú leírása⟩ |
⟨kötött formátumú erdőleírás⟩
⟨fa 6 szintjének kötött formátumú leírása⟩

⟨szögpontleírás⟩ ::= ⟨megnevezetlen szögpont leírása⟩ |
⟨megnevezett szögpont leírása⟩

⟨megnevezetlen szögpont leírása⟩ ::= | ⟨attributumhalmaz⟩

⟨attributumhalmaz⟩ ::= ⟨számhalmaz⟩

⟨megnevezett szögpont leírása⟩ ::= ⟨szögpontnév⟩⟨megnevezetlen szögpont leírása⟩

⟨szögpontnév⟩ ::= ⟨név⟩

4.2.1.1.1 ZÁRÓJELES ERDŐLEIRÁS

A LIDI-72 bemenő nyelvében az erdők leírásának legáltalánosabb formája a zárójeles erdőleírás. A zárójeles formátum a következő definíciók alapján írható le:

Erdőnek nevezzük szögpontok véges halmazát, ha vagy üres, vagy létezik az elemeinek olyan osztályozása, amely mellett az egy osztályba tartozó szögpontok halmaza fa – e fákat az erdőt alkotó fáknak nevezzük.

Szögpontok véges, nem üres halmaza fát alkot, ha létezik a halmaznak pontosan egy – a fa gyökerének nevezett – olyan eleme, amelynek a halmazra vonatkozó komplementere erdő – ezen

erdőt a fa gyökeréhez tartozó erdőnek nevezzük.

E rekurzív definíció alapján az erdők zárójeles formátumú leírását is rekurzív módon adhatjuk meg:

- a. ha az erdő üres, zárójeles leírása üres;
- b. ha az erdő nem üres, zárójeles leírása az őt alkotó fák zárójeles leírásainak sorozata;
- c. fa zárójeles leírása nyitó- és végzárójel között a fa gyökerének leírását követően a gyökérhez tartozó erdő zárójeles leírása.

Ilyen módon az erdők teljesen zárójelezett leírását kapjuk. A LIDI-72 bemenő nyelve lehetővé teszi a leírás olyan egyszerűsítését, amely mellett egy végzárójel és az őt közvetlenül követő nyitózárlójel vesszővel helyettesíthető. Ez az egyszerűsített zárójeles leírás a következőképpen adható meg pontosan:

- a. ha az erdő üres, zárójeles leírása üres;
- b. ha az erdő nem üres, zárójeles leírása nyitó- és végzárójel között az őt alkotó fák zárójeles leírásainak egymástól vesszővel elválasztott sorozata;
- c. fa zárójeles leírása a fa gyökerének leírását követően a gyökérhez tartozó erdő zárójeles leírása.

A LIDI-72 bemenő nyelvében megengedett egy erdő leírása során a teljesen zárójelezett és az egyszerűsített zárójeles formátum vegyes alkalmazása.

Zárójeles formátumban – természetesen – leírhatók tetszésszerinti szintszámú fákból álló erdők, amelyek szögpontjai között megnevezetlen szögpontok is szerepelhetnek. Zárójeles formátumba bármely alapformátum skatulyázható, azaz megengedett bármely fa olyan zárójeles leírása, amelyben a gyökérhez tartozó erdő zárójelestől különböző alapformátumokban van leírva. Megengedett továbbá zárójeles formátumba kiegészítő formátumú információ leírás beskatulyázása is.

A zárójeles erdőleírás szintaktikai szabályai tehát:

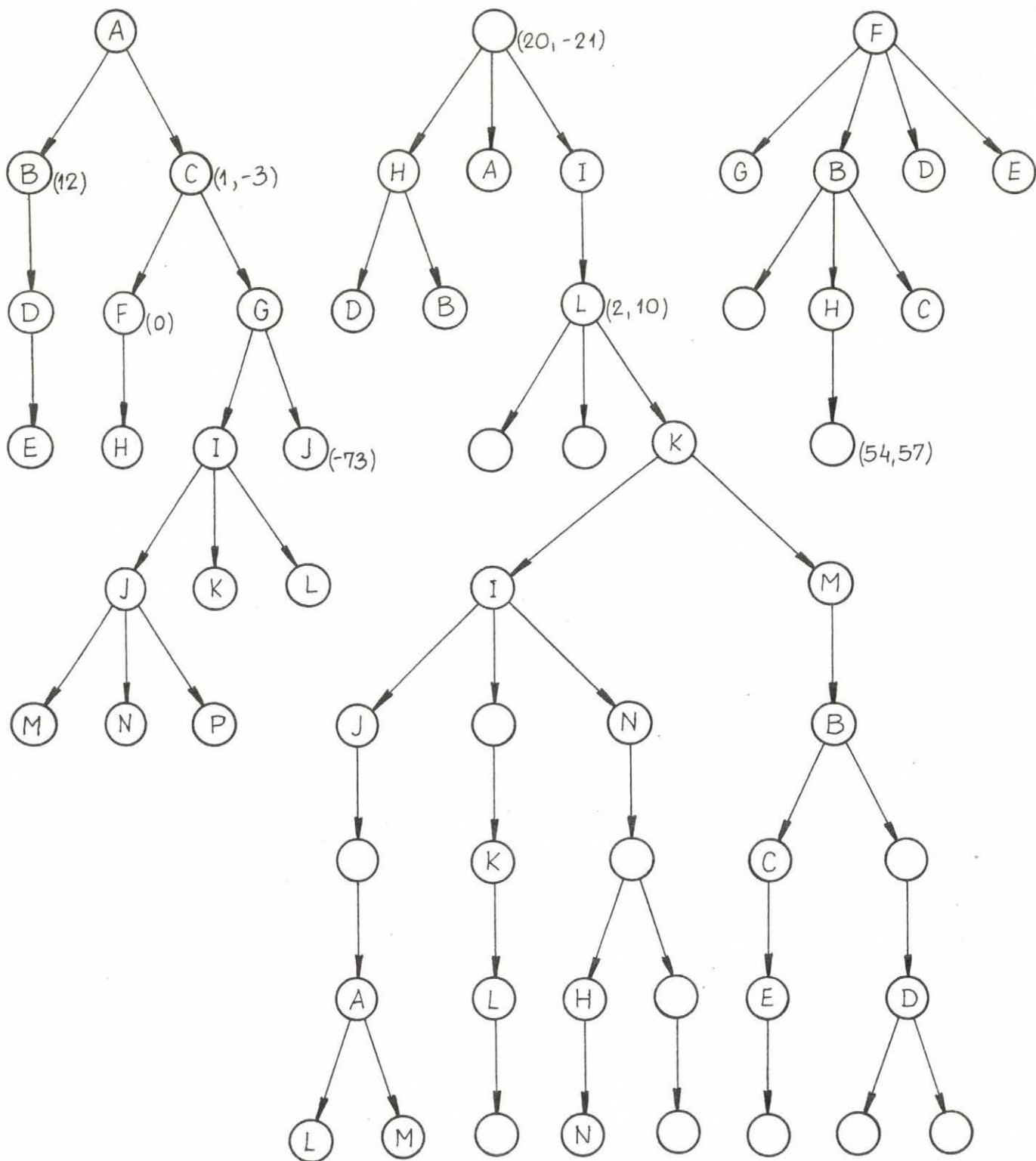
$$\langle \text{zárójeles erdőleírás} \rangle ::= | \langle \text{teljesen zárójelezett erdőleírás} \rangle | \\ \langle \text{egyszerűsített zárójeles erdőleírás} \rangle | \\ \langle \text{zárójeles erdőleírás} \rangle \\ \langle \text{teljesen zárójelezett erdőleírás} \rangle | \\ \langle \text{zárójeles erdőleírás} \rangle \\ \langle \text{egyszerűsített zárójeles erdőleírás} \rangle$$
$$\langle \text{teljesen zárójelezett erdőleírás} \rangle ::= \langle \text{fa teljesen zárójelezett leírása} \rangle | \\ \langle \text{teljesen zárójelezett erdőleírás} \rangle \\ \langle \text{fa teljesen zárójelezett leírása} \rangle$$
$$\langle \text{fa teljesen zárójelezett leírása} \rangle ::= \langle \text{kezdőzárlójel} \rangle \langle \text{fa általános leírása} \rangle \langle \text{végzárójel} \rangle$$
$$\langle \text{fa általános leírása} \rangle ::= \langle \text{a gyökér leírása} \rangle \langle \text{a gyökérhez tartozó erdő általános leírása} \rangle$$
$$\langle \text{a gyökér leírása} \rangle ::= \langle \text{szögpontleírás} \rangle$$

$\langle \text{a gyökérhez tartozó erdő általános leírása} \rangle ::= \langle \text{szabad formátumú erdőleírás} \rangle |$
 $\langle \text{kötött formátumú erdőleírás} \rangle |$
 $\langle \text{kiegészítő formátumú információ leírás} \rangle |$
 $\langle \text{a gyökérhez tartozó erdő általános leírása} \rangle$
 $\langle \text{szabad formátumú erdőleírás} \rangle |$
 $\langle \text{a gyökérhez tartozó erdő általános leírása} \rangle$
 $\langle \text{kötött formátumú erdőleírás} \rangle |$
 $\langle \text{a gyökérhez tartozó erdő általános leírása} \rangle$
 $\langle \text{kiegészítő formátumú információ leírás} \rangle$

$\langle \text{egyszerűsített zárójeles erdőleírás} \rangle ::= \langle \text{kezdőzárójel} \rangle \langle \text{általános faleírások sorozata} \rangle \langle \text{végzárójel} \rangle$

$\langle \text{általános faleírások sorozata} \rangle ::= \langle \text{fa általános leírása} \rangle | \langle \text{általános faleírások sorozata} \rangle$
 $\langle \text{vessző} \rangle \langle \text{fa általános leírása} \rangle$

Példaként megadjuk az 5. ábrán szemléltetett erdő egy zárójeles formátumú leírását (5.1 ábra):



5. ábra

ZÁRÓJELES ERDŐLEÍRÁS

MTA
SZTAKI

1.

1	6		72
	(A	(B(=A=12=) (D (E)),
	1		(C(=1, -3=) (F(=0=) (H),
	2		G (I (J(M,
	3		N,
	4		P),
	5		K,
	6		L)
	7		J(=-73=))))))
	8	((=20, -21=) (H	(D)
	9		(B))
	0		(A)
	1		(I (L(=2x10=) ()
	2		()
	3		(K (I(J((A(L,
	4		M))),
	5		(K(L())),
	6		N((H(N))
	7		(()))),
	8		M(B(C(E(),
	9		(D(,

4.2.1.1.2 LÉPTETŐJELES ERDŐLEIRÁS

Egy erdő léptetőjeles leírása az erdő teljesen zárójelezett leírása olyan átjelölésének tekinthető, amely mellett a kezdőzárójelnek – más elhatároló hiánya esetén – az *üresjel*, a végzárójelnek pedig a csillag léptetőjel felel meg. Mivel a léptetőjeles formátumban is célszerű megengedni a hatástalan *üresjelek* alkalmazását, léptetőjeles formátumban nem írhatók le olyan erdők, amelyek szögpontjai között megnevezetlen szögpontok is szerepelnek.

A léptetőjeles erdőleírás szintaktikai szabályai tehát:

⟨léptetőjeles erdőleírás⟩ :: = | ⟨léptetőjeles faleírás⟩ | ⟨léptetőjeles erdőleírás⟩⟨léptetőjeles faleírás⟩

⟨léptetőjeles faleírás⟩ :: = *üresjel* ⟨megnevezett gyökerű fa leírása⟩⟨léptetőjel⟩

⟨megnevezett gyökerű fa leírása⟩ :: = ⟨megnevezett gyökér leírása⟩

⟨a gyökérhez tartozó erdő általános leírása⟩

⟨megnevezett gyökér leírása⟩ :: = ⟨megnevezett szögpont leírása⟩

⟨léptetőjel⟩ :: = ⟨csillag⟩

Megjegyezzük, hogy a bemenő nyelvben megengedett a léptetőjeles felírás szintaxisa által kötelezően előírt *üresjel* elhagyása, ha a fa gyökerének leírását egyéb elhatároló (*1.oszlop*, végzárójel, léptetőjel vagy összetett végzárójel) előzi meg.

Példaként megadjuk az 5. ábrán szemléltetett erdő olyan leírását, amely mellett az első és a harmadik fa léptetőjeles formátumban van leírva. A harmadik fa leírása során elhagytuk a léptetőjelek miatt a fenti megjegyzés értelmében felesleges *üresjeleket*. Az 5. ábrán példaként szereplő erdő második fája – mivel gyökere megnevezetlen szögpont – nem írható le léptetőjeles formátumban, ezért leírása zárójeles formátumban adott. (5.2 ábra)

4.2.1.1.3 NÉGY SZINT KÖTÖTT FORMÁTUMÚ LEIRÁSA

A szabad formátumok a tetszésszerinti szintszámú fák leírását azáltal teszik lehetővé, hogy a következő szintre lépést bármely szinten azonos elhatároló (a kezdőzárójel, illetve az *üresjel*) jelzi, és hogy bármely szintről egy elhatároló (amely a szintszámától függetlenül a végzárójel, illetve a csillag) egyetlen szinttel történő visszalépést jelez. Az elhatárolóknak ez a szintektől független használata azonban – mint az a szemléltető példákból is látható – nagy hibalehetőséget rejt magában. Ezért kerültek kidolgozásra a kötött formátumok, amelyek a leggyakrabban előforduló (pl. a rajzok leírásában szereplő) szerkezetű fák kényelmes leírását teszik lehetővé. E fák speciális szerkezetét az jellemzi, hogy terminális szögpontok csak a legmélyebb – a negyedik, illetve a hatodik – szinten szerepelnek. Ilyen fák áttekinthető leírása lehetséges négy, illetve hat – lehetőleg különböző – előreléptető és ugyanannyi visszaléptető elhatároló birtokában, ha megköveteljük, hogy meghatározott szintről a következő szintre lépést meghatározott előreléptető, valamint a legmélyebb szintről meghatározott számú (tehát meghatározott szintre történő) visszalépést meghatározott visszaléptető elhatároló jelöljön.

Mint láttuk, szabad formátumú leírásoknál bármely szinten megengedett a kötött formátumra való áttérés; kötött formátumú leírásoknál azonban csak a legmélyebb kötött (a negyedik, illetve a hatodik) szint alatti szintek leírása történhet más – szabad – formátumban. Kötött formátumú leírásban kötött formátumú leírás még a legmélyebb szinten sem skatulyázható. Kötött formátumba nem skatulyázható kiegészítő formátumú információ leírása sem.

A négy szint kötött formátumú leírásánál a következő szintre lépést jelző elhatárolók:

1.oszlop, vagy ha a gyökér attributum nélkül megnevezetlen szögpont, akkor *7.oszlop* – a faleírás kezdete;

(egyéb elhatároló hiányában) *üresjel* – az első szintről a másodikra való áttérés;

egyenlőségjel – áttérés a második szintről a harmadik szintre;

kettőspont – a harmadik szintről a negyedik szintre való áttérés.

Visszaléptető elhatárolók csak a negyedik szinten alkalmazhatók – ezen elhatárolók a következők:

vessző – egy szinttel (a harmadik szintre) történő visszalépés, majd előrelépés egy szinttel [tehát ekvivalens a $)$ (kombinációval)];

pluszjel – két szinttel (a második szintre) történő visszalépés, majd előrelépés egy szinttel [tehát ekvivalens a $))$ (kombinációval)];

7.oszlop – három szinttel (az első szintre) történő visszalépés, majd előrelépés egy szinttel [tehát ekvivalens a $)))$ (kombinációval)];

1.oszlop – négy szinttel történő visszalépés, azaz a fa leírásának vége (amely egyúttal a következő kötött, vagy szabad formátumú információleírás, vagy a nyelv egyéb mondatának kezdete is).

A 7.oszlop elhatárolónak – a hat szint kötött formátumú leírásánál is szereplő – kettős jelentéséből következik, hogy kötött formátumú faleírást nem követhet közvetlenül olyan fa kötött formátumú leírása, amelynek gyökere attributum nélküli megnevezetlen szögpontról.

A négy szint kötött formátumú leírásának szintaktikai szabályai:

- $\langle \text{fa 4 szintjének kötött formátumú leírása} \rangle ::= \langle 1.\text{oszlop (jelzett gyökér leírása)} \rangle$
 $\langle 3 \text{ szintű erdő 4-kötött formátumú leírása} \rangle |$
 $\langle 7.\text{oszlop} \rangle \langle 3 \text{ szintű erdő 4-kötött formátumú leírása} \rangle$
- $\langle \text{jelzett gyökér leírása} \rangle ::= \langle \text{megnevezett szögpontról leírása} \rangle | \langle \text{attributumhalmaz} \rangle$
- $\langle 3 \text{ szintű erdő 4-kötött formátumú leírása} \rangle ::= \langle 3 \text{ szintű fa 4-kötött formátumú leírása} \rangle |$
 $\langle 3 \text{ szintű erdő 4-kötött formátumú leírása} \rangle$
 $\langle 7.\text{oszlop} \rangle \langle 3 \text{ szintű fa 4-kötött formátumú leírása} \rangle$
- $\langle 3 \text{ szintű fa 4-kötött formátumú leírása} \rangle ::= \langle \text{a gyökér leírása} \rangle \langle \text{egyenlőségjel} \rangle$
 $\langle 2 \text{ szintű erdő 4-kötött formátumú leírása} \rangle$
- $\langle \text{a gyökér leírása} \rangle ::= \langle \text{szögpontról leírása} \rangle$
- $\langle 2 \text{ szintű erdő 4-kötött formátumú leírása} \rangle ::= \langle 2 \text{ szintű fa 4-kötött formátumú leírása} \rangle |$
 $\langle 2 \text{ szintű erdő 4-kötött formátumú leírása} \rangle$
 $\langle \text{pluszjel} \rangle \langle 2 \text{ szintű fa 4-kötött formátumú leírása} \rangle$
- $\langle 2 \text{ szintű fa 4-kötött formátumú leírása} \rangle ::= \langle \text{a gyökér leírása} \rangle \langle \text{kettőspontról} \rangle$
 $\langle 1 \text{ szintű erdő 4-kötött formátumú leírása} \rangle$
- $\langle 1 \text{ szintű erdő 4-kötött formátumú leírása} \rangle ::= \langle 1 \text{ szintű fa 4-kötött formátumú leírása} \rangle |$
 $\langle 1 \text{ szintű erdő 4-kötött formátumú leírása} \rangle$
 $\langle \text{vessző} \rangle \langle 1 \text{ szintű fa 4-kötött formátumú leírása} \rangle$
- $\langle 1 \text{ szintű fa kötött formátumú leírása} \rangle ::= \langle \text{a gyökér leírása} \rangle$
 $\langle \text{a gyökérhez tartozó erdő szabad formátumú leírása} \rangle$
- $\langle \text{a gyökérhez tartozó erdő szabad formátumú leírása} \rangle ::= \langle \text{szabad formátumú erdőleírás} \rangle$

Példaként megadjuk az 5. ábrán szemléltetett erdő első fája négy szintjének kötött formátumú leírását. Hasonlóan megadjuk a második fa legmélyebb négy szintjéből álló erdő kötött formátumú leírását, kivéve ezen erdő második fájáét, amely a 7.oszlop kettős szerepe miatt nem írható le négy szintű kötött formátumban. A harmadik fára a kötött formátumú leírás nem alkalmazható, mivel e fának nem választható ki négy szintje úgy, hogy ne lennének magasabb szinten terminális szögpontjai. (5.3 ábra)

FA 4 SZINTJÉNEK KÖTÖTT
FORMÁTUMÚ LEÍRÁSA

M T A
SZTAKI

1	6		72
A	B (= 12 =)	= D	: E
	C (= 1, -3 =)	= F (= 0 =)	: H +
F		G	: I J (M, N, P) * K * L *
F			J (= -73 =)
((= 20, -21 =)	(H(D, B), A, I(L(= 2, 10 =)	(, K(I	
J		= A	: L,
F			M
((K(L(
N		= H	: N+
F			:
, M			
B	C	= E	:
		= D	: ,
)))) , F(G, B(, H(, C), D, E))			

5.3 ábra

4.2.1.1.4 HAT SZINT KÖTÖTT FORMÁTUMÚ LEIRÁSA

A hat szint kötött formátumú leírásánál a következő elhatárolók jelzik az eggyel mélyebb szintre történő áttérést:

1.oszlop, vagy ha a gyökér attributum nélküli megnevezetlen szögpont, akkor *7.oszlop* – a fa leírás kezdete;

(egyéb elhatároló hiányában) *üresjel* – az első szintről a másodikra való áttérés;

egyenlőségjel – áttérés a második szintről a harmadik szintre;

törtvonal – áttérés a harmadik szintről a negyedik szintre;

kettőspont – a negyedik szintről az ötödik szintre való áttérés;

törtvonal – áttérés az ötödik szintről a hatodik szintre.

Visszaléptető elhatárolók csak a hatodik szinten alkalmazhatók – ezen elhatárolók a következők:

csillag – egy szinttel történő visszalépés, majd előrelépés
[tehát ekvivalens a) (kombinációval];

vessző – visszalépés két szinttel, majd egy előrelépés
[tehát ekvivalens a)) (kombinációval];

mínuszjel – három szinttel történő visszalépés, majd egy előrelépés
[tehát ekvivalens a))) (kombinációval];

pluszjel – négy szinttel történő visszalépés, majd egy előrelépés
[tehát ekvivalens a)))) (kombinációval];

7.oszlop – visszalépés öt szinttel, majd egy előrelépés
[tehát ekvivalens a))))) (kombinációval];

1.oszlop – hat szinttel történő visszalépés, azaz a fa leírásának vége (amely egyúttal a következő kötött, vagy szabad formátumú információleírás, vagy a nyelv egyéb mondatának kezdete is).

A hat szint kötött formátumú leírásának szintaktikai szabályai:

⟨fa 6 szintjének kötött formátumú leírása⟩ :: = *1.oszlop* ⟨jelzett gyökér leírása⟩

⟨5 szintű erdő 6-kötött formátumú leírása⟩ |

⟨7.oszlop⟩⟨5 szintű erdő 6-kötött formátumú leírása⟩

⟨5 szintű erdő 6-kötött formátumú leírása⟩ :: = ⟨5 szintű fa 6-kötött formátumú leírása⟩ |

⟨5 szintű erdő 6-kötött formátumú leírása⟩

⟨7.oszlop⟩⟨5 szintű fa 6-kötött formátumú leírása⟩

⟨5 szintű fa 6-kötött formátumú leírása⟩ :: = ⟨a gyökér leírása⟩⟨egyenlőségjel⟩

⟨4 szintű erdő 6-kötött formátumú leírása⟩

- $\langle 4 \text{ szintű erdő 6-kötött formátumú leírása} \rangle ::= \langle 4 \text{ szintű fa 6-kötött formátumú leírása} \rangle |$
 $\langle 4 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
 $\langle \text{pluszjel} \rangle \langle 4 \text{ szintű fa 6-kötött formátumú leírása} \rangle$
- $\langle 4 \text{ szintű fa 6-kötött formátumú leírása} \rangle ::= \langle a \text{ gyökér leírása} \rangle \langle \text{törtvonal} \rangle$
 $\langle 3 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
- $\langle 3 \text{ szintű erdő 6-kötött formátumú leírása} \rangle ::= \langle 3 \text{ szintű fa 6-kötött formátumú leírása} \rangle |$
 $\langle 3 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
 $\langle \text{mínuszjel} \rangle \langle 3 \text{ szintű fa 6-kötött formátumú leírása} \rangle$
- $\langle 3 \text{ szintű fa 6-kötött formátumú leírása} \rangle ::= \langle a \text{ gyökér leírása} \rangle \langle \text{kettőspont} \rangle$
 $\langle 2 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
- $\langle 2 \text{ szintű erdő 6-kötött formátumú leírása} \rangle ::= \langle 2 \text{ szintű fa 6-kötött formátumú leírása} \rangle |$
 $\langle 2 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
 $\langle \text{vessző} \rangle \langle 2 \text{ szintű fa 6-kötött formátumú leírása} \rangle$
- $\langle 2 \text{ szintű fa 6-kötött formátumú leírása} \rangle ::= \langle a \text{ gyökér leírása} \rangle \langle \text{törtvonal} \rangle$
 $\langle 1 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
- $\langle 1 \text{ szintű erdő 6-kötött formátumú leírása} \rangle ::= \langle 1 \text{ szintű fa 6-kötött formátumú leírása} \rangle |$
 $\langle 1 \text{ szintű erdő 6-kötött formátumú leírása} \rangle$
 $\langle \text{csillag} \rangle \langle 1 \text{ szintű fa kötött formátumú leírása} \rangle$

Példaként megadjuk az 5. ábrán szereplő erdő második fája legmélyebb hat szintjén álló fa kötött formátumú leírását. (5.4 ábra)

FA 6 SZINTJÉNEK KÖTÖTT
FORMÁTUMÚ LEÍRÁSA

M T A
SZTAKI

1	6		72
((=20,-21=)(H(D,B),A,I(L(=2,10=)(,))	
K	I =		
F	J / : A / L*		
F	M+		
F	/ K : L / +		
F	N / : H / L,		
F			
	M-B/C : E/-		
F	: D/*		
)))			

4.2.1.2 KIEGÉSZÍTŐ FORMÁTUM

A kiegészítő formátum lehetővé teszi az olyan információ leírását, amely az alapértelmezéstől eltérő értelmezést kíván. Ez az információ a LIDI-72 bemenő nyelve szempontjából olyan szöveg, amely neveknek és (összetett kezdő- és végzárójelen belül előforduló) számoknak, valamint a vessző, törtvonal, kettőspont, kezdőzárójel, végzárójel, pluszjel, mínuszjel, egyenlőségjel, csillag, százalékjel, 7.oszlop és az *üresjel* szöveghatárolóknak egyetlen (esetleg folytatásjelekkel meghosszabított) sorba írt sorozata. A kiegészítő formátum kezdetét egy speciális *1.oszlop %* szövegkezdet, végét pedig az előírás következő mondatának kezdetét jelentő *1.oszlop* jelzi. Kiegészítő formátumú információ leírásba – természetesen – semmilyen információ leírás nem skatulyázható.

A kiegészítő formátumú információ leírás szintaktikai szabályai:

⟨kiegészítő formátumú információ leírása⟩ ::= ⟨szövegkezdet⟩⟨szöveg⟩

⟨szövegkezdet⟩ ::= *1.oszlop* ⟨százalékjel⟩

⟨szöveg⟩ ::= | ⟨szöveg⟩⟨szövegelem⟩

⟨szövegelem⟩ ::= ⟨név⟩ | ⟨szöveghatároló⟩ | ⟨számhalmaz⟩

⟨szöveghatároló⟩ ::= ⟨vessző⟩ | ⟨törtvonal⟩ | ⟨kettőspont⟩ | ⟨kezdőzárójel⟩ | ⟨végzárójel⟩ | ⟨pluszjel⟩ | ⟨mínuszjel⟩ | ⟨egyenlőségjel⟩ | ⟨csillag⟩ | ⟨százalékjel⟩ |
 ⟨7.oszlop⟩ | *üresjel*

A kiegészítő formátumban leírt információ egyetlen megnevezetlen terminális elem elhelyezését eredményezi, amelynek attributumai a szöveg elemeihez rendelt értékek az elemek szövegben való előfordulásának sorrendjében.

A szövegben előforduló névnek megfelelő attributum a névhez rendelt (nemnegatív) belső név, a szöveghatárolóknak megfelelő attributumértékek az elhatárolók belső kódjai negatív előjellel. A szöveg egy elemeként szereplő k elemű számhalmaznak megfelelő $k + 2$ attributum: a számhalmaz összetett kezdőzárójelének negatív előjellel vett belső kódját követi a számhalmaz elemeinek száma, majd a halmazt alkotó számok értékei következnek.

Összefoglalva:

Szövegelem	Attributum
név	belső név
7.oszlop	– 2
kettőspont	– 3
egyenlőségjel	– 4
pluszjel	– 5
végzárójel	– 6
mínuszjel	– 7
csillag	– 8

Szövegelem	Attributum
üresjel	– 9
törtvonal	– 10
vessző	– 11
kezdőzárójel	– 12
százalékjel	– 15
számhalmaz	– 13, az elemek száma, a számok értékei.

Példaként megadunk egy kiegészítő formátumú információ leírást, amelynek hatására elhelyezkedő elem képét a 6. ábrán szemléltetjük. (6.1 ábra)

	-14
	6
	5
	-13
	4
	-13
nb(Y)	
	-2
	4
	-3
	-2
	1
	4
	-13
nb(X)	
	-2
	2
	2
	2
	-13
nb(TIPUS)	
	-2
	-6
nb(Y)	
	-7
nb(X)	
	-12
	-8
nb(2)	
	-4
	-6
nb(Y)	
	-11
nb(X)	
	-12
nb(F2)	
	-2
nb(Y)	
	-5
nb(X)	
	-4
	-6
nb(Y)	
	-11
nb(X)	
	-12
nb(F1)	
	47
TA	

6. ábra

KIEGÉSZÍTŐ FORMÁTUMÚ
INFORMÁCIÓ LEÍRÁS

M T A
SZTAKI

1	6				72
%		F1(X,Y)=X+Y			
		F2(X,Y)=2*(X-Y)			
		TIPUS(=2*2=)			
		X(=			
	C	1	-2		
	C	-3	4		
	C	=)			
		Y(=			
	C	-13	5		
	C	6	-14		
	C	=)			

4.2.2 ÚTELŐIRÁS

A bemenő nyelv valamely előírásában egy információ leírással adott erdő alapértelmezés szerint a lista bemenő eleme alatti *INFO* nevű szögpontra kerül bekapcsolásra. Ezen alapértelmezéstől való eltérés az *1.oszlop PATH* kezdetű, (esetleg folytatásjelek segítségével meghosszabbított) egyetlen sorba írt útelőírásokkal érhető el: egy útelőírás hatására az előírásban az adott útelőírást követő információ leírással adott erdő az útelőírással meghatározott út végpontját jelentő szögpontra alá kapcsolódik be. Egy útelőírás egymástól 7.oszlop jellel elválasztott útkijelölések sorozata; az információ bekapcsolása szempontjából az útelőírás által meghatározott út végpontját az útelőírásban utolsóként szereplő útkijelölés határozza meg.

Egy útra való későbbi hivatkozás az út nevével történhet, amelyet útdefinícióval rendelünk az úthoz. Egy útdefinícióban az út leírását az útnév után, attól egyenlőségjellel elválasztva adjuk meg.

Valamely kiindulási szögpontról a kijelölendő végpontig vezető út leírása az utat alkotó útszakaszok leírásainak sorozata. Legegyszerűbb a meghatározott útszakasz leírása, amelynél az előző útszakasz végpontjától kiindulva a leírandó úton fekvő minden szögpontra nevet megadjuk egymástól vesszővel elválasztva. A meghatározott útszakaszok leírásait vesszővel kapcsoljuk a nem teljesen meghatározott útszakaszok leírásai után, amelyeknek a következő két formáját különböztetjük meg. Egy meghatározott $k + 1$ ($k \geq 0$) hosszúságú útszakasz ($= k = n$) alakú leírása az előző útszakasz végpontjától számított $k + 1$. szinten lévő, n nevű szögpontra jelöli ki a szakasz végpontjaként (tehát k az útszakasz fel nem sorolt szögpontjainak száma). Valamely $(= =)n$ alakú leírással adott meghatározatlan útszakasz végpontja az előző útszakasz végpontjától elérhető n nevű szögpontra.

Az útszakaszok leírásának egyszerű sorozatával kijelölt utak kiindulási pontjaként a lista bemenő eleme alatti *INFO* nevű szögpontra szerepel. Kiindulási pontként kijelölhető valamely előzőleg definiált út végpontja is úgy, hogy a már definiált út nevére követő törtvonal után adjuk meg az (ezen út végpontjától a végpontként kijelölendő szögpontra vezető valamely) út leírását. Megjegyezzük, hogy a megnevezetlen út végpontjaként a lista bemenő eleme alatti *WORK* szögpontra szerepel, az üres útkijelölés pedig az *INFO* nevű szögpontra adja az út végpontjaként.

Az útelőírások szintaktikus szabályai:

$\langle \text{útelőírás} \rangle ::= \langle \text{útelőírás kezdete} \rangle \langle \text{útkijelölés} \rangle \mid \langle \text{útelőírás} \rangle \langle 7.\text{oszlop} \rangle \langle \text{útkijelölés} \rangle$

$\langle \text{útelőírás kezdete} \rangle ::= 1.\text{oszlop} \text{ PATH}$

$\langle \text{útkijelölés} \rangle ::= \mid \langle \text{útleírás} \rangle \mid \langle \text{útdefiníció} \rangle$

$\langle \text{útleírás} \rangle ::= \langle \text{útkezdete kijelölése} \rangle \mid \langle \text{útszakaszleírás} \rangle \mid \langle \text{útleírás} \rangle \langle \text{vessző} \rangle \langle \text{meghatározott szakasz leírása} \rangle \mid \langle \text{útleírás} \rangle \langle \text{nem teljesen meghatározott szakasz leírása} \rangle$

$\langle \text{útkezdete kijelölése} \rangle ::= \langle \text{útnév} \rangle \langle \text{törtvonal} \rangle$

$\langle \text{útnév} \rangle ::= \mid \langle \text{név} \rangle$

$\langle \text{útszakasz leírás} \rangle ::= \langle \text{meghatározott szakasz leírása} \rangle |$
 $\langle \text{nem teljesen meghatározott szakasz leírása} \rangle$
 $\langle \text{meghatározott szakasz leírása} \rangle ::= \langle \text{szögpontnév} \rangle |$
 $\langle \text{meghatározott szakasz leírása} \rangle \langle \text{vessző} \rangle \langle \text{szögpontnév} \rangle$
 $\langle \text{nem teljesen meghatározott szakasz leírása} \rangle ::= \langle \text{meghatározott hosszúságú szakasz leírása} \rangle |$
 $\langle \text{meghatározatlan szakasz leírása} \rangle$
 $\langle \text{meghatározott hosszúságú szakasz leírása} \rangle ::= \langle \text{összetett kezdőzárójel} \rangle \langle \text{nemnegatív egész szám} \rangle$
 $\langle \text{összetett végzárójel} \rangle \langle \text{szögpontnév} \rangle$
 $\langle \text{nemnegatív egész szám} \rangle ::= \langle \text{természetes szám} \rangle | \langle \text{pluszjel} \rangle \langle \text{természetes szám} \rangle$
 $\langle \text{meghatározatlan szakasz leírása} \rangle ::= \langle \text{összetett kezdőzárójel} \rangle \langle \text{összetett végzárójel} \rangle \langle \text{szögpontnév} \rangle$
 $\langle \text{útdefiníció} \rangle ::= \langle \text{útnév} \rangle \langle \text{egyenlőségjel} \rangle \langle \text{útleírás} \rangle$

Példaként megadunk egy útleírásokat is tartalmazó (nagyon gazdaságtalan bevittelt adó) előírást, és a 7. ábrán megmutatjuk a hatására elhelyezkedő információ gráfját, ahol kettős körrel az előírás értelmezése előtt bentlévő szögpontokat jelöljük.

4.2.3 ELJÁRÁSELŐÍRÁS

A LIDI bemenő nyelvének 1972. évi változatában az eljáráselőírások szintaxisa és szemantikája még nincs pontosan meghatározva, csak az rögzített, hogy kezdetük *1.oszlop PROC* alakú.

4.2.4 MEGJEGYZÉSEK

A LIDI-72 bemenő nyelvében bármely elhatároló után írhatók *1.oszlop*-kombinációval kezdődő, egyetlen soros megjegyzések. Minden megjegyzés legfeljebb 71, *1.oszloptól* különböző alapjel tetszőszerinti sorozata (amely tehát nem tartalmazhat se *7.oszlop* elhatárolót, se folytatásjelet); a megjegyzésekben szereplő alapjelek hatástalanok.

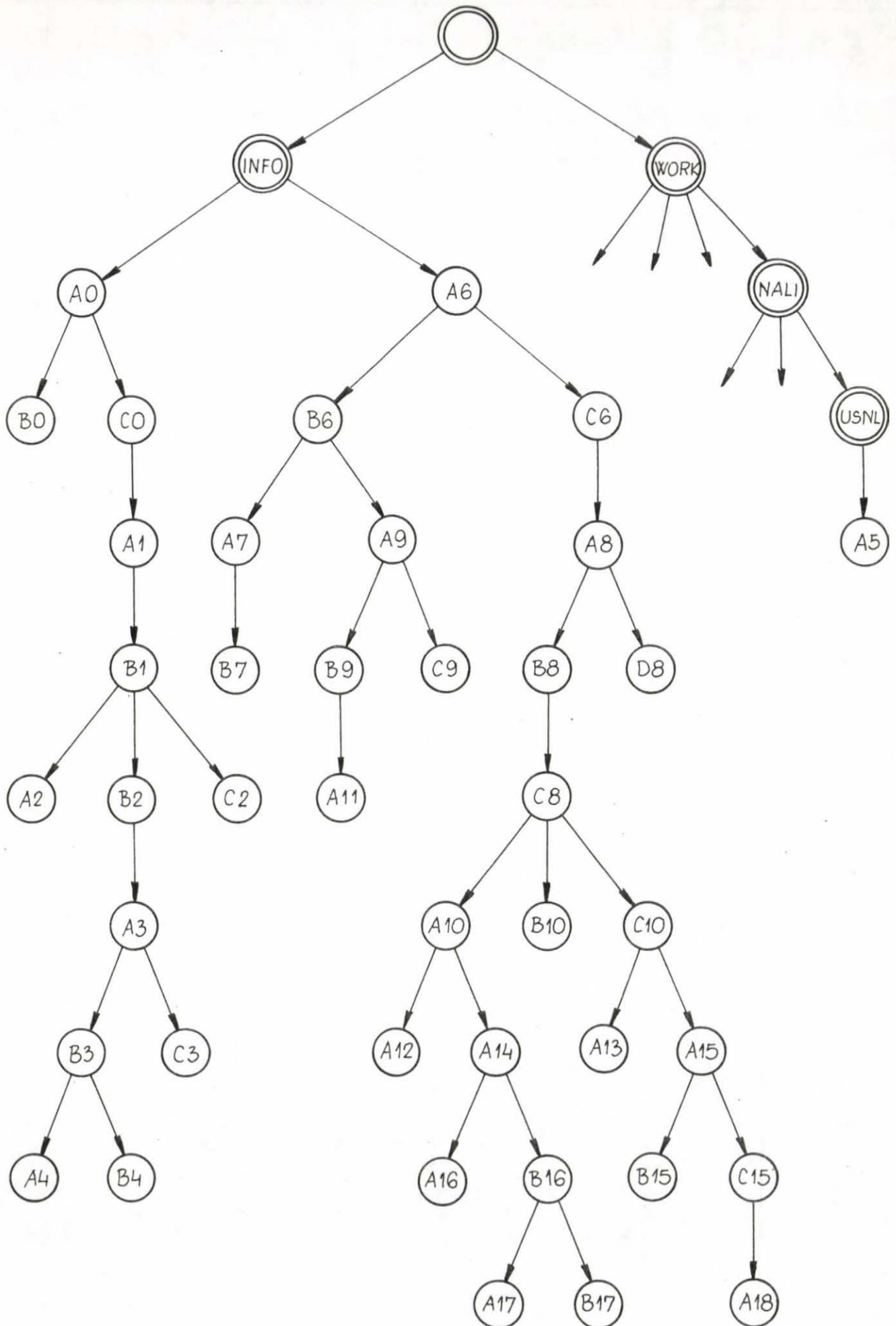
A megjegyzésekben szereplő alapjelek számára vonatkozó korlátozás figyelmen kívül hagyásával a megjegyzések szintaktikai szabályai:

$\langle \text{megjegyzés} \rangle ::= \langle \text{megjegyzés kezdete} \rangle | \langle \text{megjegyzés} \rangle \langle \text{1.oszloptól különböző alapjel} \rangle$

$\langle \text{megjegyzés kezdete} \rangle ::= \text{1.oszlop} -$

$\langle \text{1. oszloptól különböző alapjel} \rangle ::= \text{üresjel} | \langle \text{betű} \rangle | \langle \text{számjegy} \rangle |$

$. | , | / | : | (|) | + | - | = | * | \%$



7. ábra

ÚTELOÍRÁS

M T A
SZTAKI

1.

1	6	72
(A0(B0,CO))		
PATH A0,CO		
A1 B1**		
PATH (=3=)B1		
A2*B2*C2*		
PATH (= =)B2		
(A3(B3,C3))		
PATH A0,CO (=2=)B2 (= =)B3		
A4*B4*		
PATH /NALI,USNL		
A5*		
PATH		
A6 B6*C6**		
PATH U2=A6,C6		
U1=(= =)B6		
A7 B7**		
PATH U2/		
(A8(B8(C8),D8))		
PATH U1/		
(A9(B9,C9))		

ÜTELOÍRÁS

M T A
SZTAKI

2.

1	6	72
PATH	U2/A8, B8, C8	
A10*	B10*C10*	
PATH	U1/(= 1=)B9	
A11*		
PATH	U1=U2/A8, B8(= =)A10	
A12*		
PATH	U2=U2/A8, B8(= 1=)C10	
(A13)		
PATH	U1/	
A14*		
PATH	U2/	
A15	B15*C15**	
PATH	U1/A14	
(A16, B16)		
PATH	U1/(= =)A14, B16	
A17*	B17	
PATH	U2/A15(=0=)C15	
A18*		
+		

7.1 ábra

5. A RENDSZER FEJLESZTÉSE

A rendszer fejlesztésével kapcsolatban kétirányú tevékenység elvégzése szükséges: egyrészt a LIDI-72 rendszerben megoldott feladatok általánosítása, másrészt a célként kitűzött, de a LIDI-72 rendszerben még ki nem dolgozott feladatok megoldása.

A LIDI-72 kidolgozása a számítógépes tervező rendszer kidolgozásával párhuzamosan folyt: a mint a belső reprezentáció rögzítésre került és a legfontosabb műveleteket realizáló szubrutinok elkészültek, a tervező rendszerben azonnal felhasználásra kerültek; a LIDI-72 rendszer további műveleteket megvalósító programjai pedig elsősorban a tervező rendszer kidolgozása során felmerült igények alapján készültek. A számítógépes tervezésben való felhasználás tapasztalatai alapján szükségesnek látszik a LIDI-72 olyan általánosítása, amelyben az attributumok bizonyos osztályainak (pl. nevek, egész számok stb.) kezelésével kapcsolatos egyes tevékenységeket a felhasználó helyett maga a rendszer lát el. A tervező rendszer működésének gyorsítása érdekében célszerűnek látszik a LIDI-72 egyes szubrutinjaiba beépített ellenőrző tevékenységek elválasztása a szubrutin által megvalósítandó műveleteket végző tevékenységektől, valamint a szabad mezők kezelésének bizonyos módosítása. A LIDI rendszer teljesebbé tétele érdekében szükséges a gráfokon végzendő műveletek körének bővítése és e műveleteket realizáló programok kidolgozása.

A LIDI-72 rendszer másik irányba történő fejlesztése során elsőrendű feladatként az output és input egységesítése szerepel. További feladat a rendszer kiterjesztése a közvetlen hozzáférésű háttértárolókra, valamint az információ katalogizálásának megoldása. Célszerűnek látszik biztosítani a felhasználó (pl. a tervező rendszer) eljárásainak automatikus végrehajtását, ami vezérlő-program kidolgozását teszi szükségessé.

IRODALOMJEGYZÉK

- [1] Mc Carthy et al: LISP 1.5 Programmer's Manual
The M.I.T. Press, 1962.
- [2] Smith, D.K.: An Introduction to the List Processing Language SLIP
Programming Systems and Languages
(ed. Rosen, S. 1967.)
- [3] Bobrow, D.G., Raphael, B.: A Comparison of List-Processing Computer Languages
Programming Systems and Languages
(ed. Rosen, S. 1967.)
- [4] Knuth, D.E.: The Art of Computer Programming. Vol. 1.
Addison-Wesley Publishing Co. 1968.

TARTALOMJEGYZÉK

1.	Bevezetés	3
2.	A belső reprezentáció	5
2.1	Az információ ábrázolása	5
2.2	Léptetőelemek	9
2.3	A szabad mezők nyilvántartása	9
3.	Műveletek a listán	12
3.1	Kezdeti állapot beállítása	12
3.2	Az információ bevitele	14
3.3	Információ bekapcsolása	14
3.4	Információ törlése	15
3.5	Szabad mezők kezelése	15
3.6	Bejárások	16
3.7	Szerkezeti vizsgálatok	17
4.	A bemenő nyelv leírása	18
4.1	Alapjelek, elhatárolók, elemek	18
4.2	Mondatok, előírás	20
4.2.1	Az információ leírása	20
4.2.1.1	Alapformátumok; szögpontok leírása	20
4.2.1.1.1	Zárójeles erdőleírás	21
4.2.1.1.2	Léptetőjeles erdőleírás	27
4.2.1.1.3	Négy szint kötött formátumú leírása	29
4.2.1.1.4	Hat szint kötött formátumú leírása	32
4.2.1.2	Kiegészítő formátum	35
4.2.2	Utelőírás	39
4.2.3	Eljáráselőírás	40
4.2.4	Megjegyzések	40
5.	A rendszer fejlesztése	44
	Irodalomjegyzék	45

A TANULMÁNYOK sorozatban eddig megjelentek:

- 1/1973 Pásztor Katalin: Módszerek Boole-függvények minimális vagy nem redundáns, $\{\wedge, \vee, \neg\}$ vagy $\{\text{NOR}\}$ vagy $\{\text{NAND}\}$ bázisbeli, zárójeles vagy zárójel nélküli formuláinak előállítására
- 2/1973 Вашкеви Иштван: Расчленение многосвязных промышленных процессов с помощью вычислительной машины
- 3/1973 Ádám György: A számítógépipar helyzete 1972 második felében
- 4/1973 Bányász Csilla: Identification in the presence of drift
- 5/1973* Gyürki J.-Laufer J.-Girnt M.-Somló J.: Optimalizáló adaptív szerszámgepirányítási rendszerek
- 6/1973 Szelke Erszébet-Tóth Károly: Felhasználói Kézikönyv (USER MANUAL) a Folytonos Rendszerek Szimulációjára készült ANDISIM programnyelvhez
- 7/1973 Legendi Tamás: A CHANGE nyelv/multiprocesszor
- 8/1973 Klafszky Emil: Geometriai programozás és néhány alkalmazása
- 9/1973 R.Narasimhan: Picture Processing Using Pax
- 10/1973 Dibuz Ágoston-Gáspár János-Várszegi Sándor: MANU-WRAP hátlaphuzalozó, MSI-TESTER integrált áramköröket mérő, TESTOMAT-C logikai hálózatokat vizsgáló berendezések ismertetése
- 11/1973 Matolcsi Tamás: Az optimum-számítás egy új módszeréről
- 12/1973 Makroprocesszorok, programozási nyelvek.
Cikkgyűjtemény az NJSzT és SzTAKI közös kiadásában.
Szerkesztette: Legendi Tamás
- 13/1973 Jedlovsky Pál: Új módszer bonyolult retifikáló oszlopok vegyész-mérnöki számítására
- 14/1973 Bakó András: MTA Kutatóintézeteinek bérszámfejtése számítógéppel
- 15/1973 Ádám György: KELET-NYUGATI KAPCSOLATOK A SZÁMÍTÓGÉPIPARBAN

*-gal jelölt kivételével a TANULMÁNYOK megrendelhetők az Intézet Könyvtáránál (Budapest, I. Uri u. 49.)

A kiadásért felelős:

Dr. Vámos Tibor

az

MTA Számítástechnikai és Automatizálási
Kutató Intézet
igazgatója

Jelen dolgozat a 3.2.4 "Logikai áramkörök
számítógépes tervezése" c. intézeti alapku-
tatási téma keretében készült.

Beérkezett: 1973. október 25.

