

Adatszerkezetek és algoritmusok

Állományok (szeriális, szekvenciális)

Dr. Fazekas Attila



DEBRECENI
EGYETEM



Állomány (file)

- Az állomány fogalma mindig valamilyen periférián értelmezhető.
- A háttértár határozza meg az állomány fizikai struktúráját.
- A valós világ modellezése (az absztrakció első szintje) ugyanúgy történik állományok esetében is, mint az adatszerkezeteknél.
- Az absztrakció második szintjén megjelenik a fizikai állomány. Tehát van egy logikai állomány és egy fizikai állomány.



A logikai állomány fogalmi rendszere

- **Adatelem, adattétel:** A logikai állomány legkisebb, önállóan értelmezhető része. Jellemzői:
 - **Név:** hasonló a szerepe, mint a változók esetén
 - **Típus:** milyen típusú értéket vehet fel
 - **Hossz:** általában bájtokban/karakterekben határozzuk meg
 - **Fix hosszúságú** (pl.: születési évszám → 4 karakter)
 - **Változó hosszúságú** (pl.: születési hely)



A logikai állomány fogalmi rendszere (folyt.)

- **Adatcsoport:** Az adatelemek önálló, névvel ellátott együttese. Az alábbi csoportokat különböztetjük meg:
 - **Vektor adatcsoport:** Különböző típusú adatelemeket fog össze valamilyen nagyobb logikai egységbe (pl.: a lakcím tartalmazza az irányítószámot, településnevet, utca nevét, házszámot).
 - **Ismétlődő adatcsoport:** Ebben az esetben olyan adatelemünk van, amelyiknek több értéke is lehet (pl.: gyermekünk neve). Az ismétlődések száma lehet 0 is.
 - **Összetett adatcsoport:** Az előző kettő kombinációja: egy vektortípusú adatcsoport, melynek egyik eleme ismétlődik.



A logikai állomány fogalmi rendszere (folyt.)

- **Logikai rekord:** Az adatelemek és adatcsoportok adott sorrendű, logikailag összetartozó együttese (Nincs önálló neve.). A logikai rekordok szerkezete alapján:
 - **Fix rekordformátum:** A logikai rekordok azonos szerkezetűek és azonos hosszúságúak. Ez eleve kizárja a változó hosszúságú adatelemeket és az ismétlődő adatcsoportokat. Ez a legegyszerűbb szerkezetű logikai rekord.
 - **Változó rekordformátum:** A logikai rekordok szerkezete azonos, de hosszuk eltérő lehet. Változó hosszúságú adatelem, vagy ismétlődő adatcsoport esetén ez rekordformátum alkalmazandó. Ez a leggyakoribb rekordformátum.



A logikai állomány fogalmi rendszere (folyt.)

- **Határozatlan rekordformátum:** A rekordok hossza és szerkezete is eltérő lehet. Adatelemek hiányozhatnak az egyes rekordokból. A fizikai szinten ezt nehéz kezelni.
- **Logikai rekordazonosító:** Azt az adatelemet, vagy az adatelemeknek azon együttesét, amelynek vagy amelyeknek értéke egyedi, azaz minden konkrét rekordban más és más, logikai rekordazonosítónak nevezzük (pl.: személyi szám). Ha ez egyetlen adatelem, akkor **egyszerű**, egyébként **összetett rekordazonosító**ról beszélünk.



A logikai állomány fogalmi rendszere (folyt.)

- **Logikai állomány:** A logikai rekordok valamilyen együttese, amelyet névvel látunk el. A logikai állomány egy absztrakt adatszerkezet. Szerkezete alapján lehet:
 - **Struktúra nélküli:** A logikai rekordok sorrendje tetszőleges.
 - **Asszociatív:** A rekordok csoportosíthatóak valamilyen egyértelmű módon. Pl.: csoportosítsunk a logikai rekordazonosítók alapján.
 - **Szekvenciális:** A logikai rekordok között valamilyen sorrend értelmezett. (pl.: azonosító szerinti rendezettség)
 - **Hiearchikus:** Különböző logikai rekordok egy állományban úgy helyezkednek el, hogy az leginkább egy fával jellemezhető.



A fizikai állomány fogalmi rendszere

- **Mező:** Kölcsönösen egyértelműen megfelelője az adatelemnek. Itt jellemzője az ábrázolási mód, hiszen megjelenik a háttértáron. A hossz itt bájtokban értendő. Nem fix hosszú adatelemek kezelésénél az adatelem mezőjét is fix hosszon kell megjeleníteni.
- **Mezőcsoport:** Megfeleltethető az adatcsoportnak.
- **Blokk:** Megfeleltethető a logikai rekordnak.
- **Fizikai állomány:** Megfeleltethető a logikai állománynak, nem más mint blokkok sorozata. Saját neve van, kezelése operációs szinten zajlik.



Blokk

- **Blokk** az a legkisebb adatmennyiség, amely **egyszerre mozog** a tár és a periféria között.
- A blokk és a logikai rekord között a megfeleltetés többféleképpen történhet: egy logikai rekord alkot egy blokkot, több logikai rekord alkot egy blokkot (**blokkolás**), egy logikai rekord több blokkban jelenik meg (**szegmentálás**).
- Egyes rendszerek a blokk méretét rögzítik, és a felhasználó nem változtathatja meg. Más rendszerek megengedik, hogy a felhasználó szabja meg a blokk méretét.
- Ha a blokk mérete rögzített, akkor elképzelhető, hogy a rendszer nagy blokkoknál automatikus szegmentálást hajt végre.



DEBRECENI
EGYETEM

Logikai rekordformátumok \Rightarrow blokk

- **Fix rekordformátum**

- Kezelése a legkönnyebb, minden rendszer tudja kezelni. A rekordhossz az állomány jellemzőjeként írható le. Ebben az esetben
 - **Egy blokk – egy rekord:** Az összes blokkjellemező ismert.
 - **Blokkolás:** Az állományra jellemző a blokkolási tényező (hány rekord kerül egy blokkba). Az utolsó blokk lehet töredék-blokk.
 - **Szegmentálás:** Meghatározandó a rekord feltördeléséhez a blokk mérete.



DEBRECENI
EGYETEM

Logikai rekordformátumok \Rightarrow blokk (folyt.)

- **Változó rekordformátum**

- **Egy blokk – egy rekord** (1 rekord-1 blokk): Egyik megoldás a blokk méretének rögzítése, amit úgy teszünk meg, hogy a leghosszabb rekord is beleférjen. A másik megoldás a változó hosszúságú blokk használata. Ekkor a blokk elején jelölni kell a blokk hosszát.
- **Blokkolás**: Egyik esetben fix a blokkméret. A rendszer a rekordokat addig pakolja a blokkba, amíg beleférnek, a megmaradt helyet feltölti. A többi rekordot a következő blokkba rakja. A másik esetben maximális blokkméret kerül megadásra. Változó méretű blokkok keletkeznek, így a blokk hosszát is tárolni kell.



Logikai rekordformátumok \Rightarrow blokk (folyt.)

- **Szegmentálás:** Problémás. A blokkoláshoz hasonlóan, plusz információt kell elhelyezni a blokkban. Rendszertől függ, hogy a plusz információt hogyan tárolja.



DEBRECENI
EGYETEM

Logikai rekordformátumok \Rightarrow blokk (folyt.)

- **Határozatlan rekordformátum.** Minden rekord tárolásánál az előzőek mellett olyan plusz információkat kell tárolni, amely az adott rekordra vonatkozik, azaz hogy a rekordnak milyen mezői hiányoznak.



DEBRECENI
EGYETEM

Műveletek (logikai rekord)

- **Létrehozás:** Az állomány egy háttértáron létrejön. Ki kell választani a fizikai állomány szervezési módját. Az adott szervezési módnak megfelelően kell a logikai rekordokat elhelyezni a fizikai állományban.
- **Bővítés:** Az állomány rekordjainak darabszáma nő, új rekord kerül be az állományba.
- **Törlés**
 - **Logikai:** Az állomány rekordjainak darabszáma nem változik, de a logikailag törölt rekordok meg vannak jelölve, a feldolgozásban nem vesznek részt.
 - **Fizikai:** Az állomány rekordjainak darabszáma csökken.



DEBRECENI
EGYETEM

Műveletek (folyt.)

- **Keresés:** Soros és szekvenciális elérésnél egy kitüntetett rekord megkeresése.
- **Rendezés:** A szokásos (kulcs alapján).
- **Feldolgozás:** Egy adott állományban lévő információhoz való hozzáférés.
- **Újraszervezés:** Az állomány, a háttértár egy másik területén történő újra létrehozása úgy, hogy közben megváltozik vagy a szerkezete, vagy a tartalma, esetleg mindkettő.



DEBRECENI
EGYETEM

Állományszerkezetek

- **Egyszerű.** Csak a logikai rekordokat tartalmazza.
 - **Szeriális**
 - **Szekvenciális**
 - **Direkt**
 - **Random**
- **Összetett.** A logikai rekord adatain túl szerkezethordozó adatok is megjelennek, amelyek magáról az állományról adnak információt és a feldolgozást segítik.



Egyszerű állományszerkezetek

- Nézzük meg a kapcsolatot a logikai rekordazonosító és a logikai rekord háttértáron elfoglalt helye között. (1)
- Nézzük meg a kapcsolatot a logikai rekordazonosítók között. (2)

	1	2
szeriális	nincs	nincs
szekvenciális	nincs	van
direkt	van	van
random	van	nincs



Szeriális állomány

- Nincs szerkezete, a rekordok sorrendje tetszőleges.
- Tetszőleges háttértáron megjelenhet, mindegyik rekordformátumot tudja kezelni, tetszőlegesen lehet blokkolni és szegmentálni.
- Egyszerűen kezelhető.
- Viszonylag gyors, kivéve ha egy adott rekordot keresek, mert a teljes keresés időigényes.
- Minden operációs rendszer, programnyelv stb. tudja kezelni.
- Alapvető szerepet játszik olyan eseteknél, amikor a rekordok **véletlenszerűen** állnak elő, vagy amikor a véletlenszerűen érkező rekordokat átmenetileg gyűjteni kell, hogy az így létrejövő állományból valamilyen szerkezettel rendelkező állományt hozzunk létre.



Szeriális állomány műveletei

- **Létrehozás:** A tetszőleges sorrendben érkező rekordok bekerülnek az állományba a beérkezési sorrend szerint.
- **Bővítés:** Nincs rendezettség, így a bővítés nagyon egyszerű, az új rekord az állomány végére íródik.
- **Törlés:** Fix rekordformátum esetén lemezes háttértárnál lehet fizikai törlés (az utolsó rekorddal felülírom a törlendőt). Szalagnál és nem fix rekordformátumnál csak logikai törlésről beszélhetünk.
- **Csere:** A lemezen elhelyezett fix rekordformátumú szeriális állomány esetén lehet cseréről beszélni.



Szeriális állomány műveletei (folyt.)

- **Elérés:** Csak soros elérésről beszélhetünk, mert csak fizikai sorrend van.
- **Keresés:** Teljes keresés.
- **Rendezés:** Nincs.
- **Feldolgozás:** Nem más, mint a rekordok végigolvasása. Viszonylag lassú.
- **Újraszervezés:** A szerkezetátalakítás miatt ez nem merül fel, mert nincs szerkezet.

Szekvenciális állomány

- A rekordok az azonosító (elsődleges kulcs) szerint rendezettek. Ez egyértelmű sorrendet és így szerkezetet biztosít.
- Létrehozható mind szalagon, mind lemezen.
- Minden rekordformátumot tud kezelni, a blokkolás is tetszőleges.
- A szakma szűkebb értelemben az azonosító szerinti növekvő sorrendben rendezett állományt nevezi szekvenciálisnak, de a csökkenő sorrendű is kielégíti a fogalmakat.
- Darabolással szintén szekvenciális állományokat kapok.



DEBRECENI
EGYETEM

Szekvenciális állomány műveletei

- **Bővítés:** Nem hajtható végre közvetlenül, csak újraszervezéssel.
- **Törlés:** Csak logikai törlés realizálható, fizikai csak újraszervezéssel.
- **Csere:** Fix rekordformátum esetén lemezen lehet, bármely más esetben csak újraszervezéssel.
- **Elérés:** A szekvenciális állománynál a logikai és a fizikai sorrend egybeesik, ezért a soros és a szekvenciális elérés ugyanazt jelenti.



Szekvenciális állomány műveletei (folyt.)

- **Keresés:** Lineáris keresés, előnyösebb ha több rekordot keresünk egyszerre.
- **Feldolgozás:** Alapja a keresés.
- **Újraszervezés:** Így oldható meg a csere, a fizikai törlés és a bővítés. Nem a szerkezet változik.



DEBRECENI
EGYETEM

Szekvenciális állomány

Előnyei

- Soros és közvetlen elérésű háttértárolón egyaránt létrehozható.
- Jó kapacitáskihasználást tesz lehetővé.
- Szekvenciális módon való feldolgozása egyszerű.

Hátrányai

- Nem támogatja a közvetlen elérést.
- Aktualizálása újrászervezéssel oldható meg.
- Létrehozását rendezés vagy összeválogatás előzi meg.



DEBRECENI
EGYETEM

Köszönöm a figyelmet!