Relációs algebra lekérdezések optimalizációja

Adatbázisok használata

Mi a cél?

- Moore-törvénye: (Gordon Moore) szerint az integrált áramkörök sok jellemzőjének fejlődése exponenciális, ezek az értékek 18 havonta duplázódnak. Ilyenek például:

  (i) processzorok sebességének és árának aránya,

  (ii) lemez egy bitre eső ára és a lemezen tárolható bájtok száma.

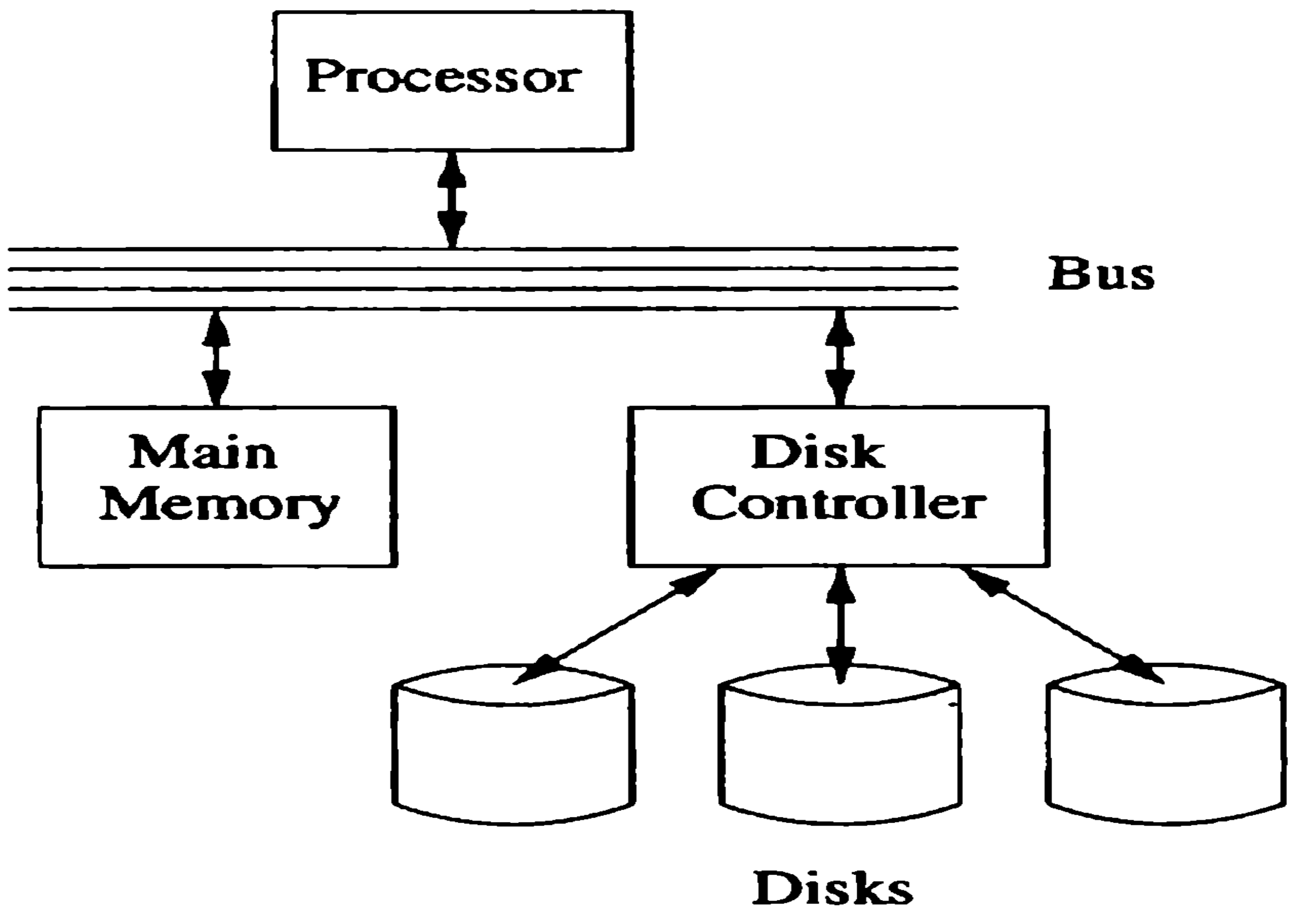
- Más paraméterek azonban sokkal lassabban fejlődnek. Ilyenek például:

  (i) központi memóriában milyen gyorsan lehet az adatokat elérni,

  (ii) az a sebesség, amellyel a lemez mozog.

- Emiatt egy-egy nagy adathalmazzal dolgozó algoritmus optimalizációjánál az a lényeges szempont, hogy a feladatot minél kevesebb adatmozgatással tudjunk megoldani a háttértároló és a központi memória között.

Számítógép rendszer sematikus ábrája



Egy lehetséges megközelítés

- Az adatbázisoknál az előbbiek nyilván úgy értendők, hogy szeretnénk minél kevesebb lemez olvasási és írási (I/O) műveletet végrehajtani egy-egy lekérdezés végrehajtása során.
- Az legegyszerűbb megközelítés, ha igyekszünk minél kisebb méretű relációkkal dolgozni.
- Az optimalizáció során relációs algebrai azonosságokat fogunk alkalmazni. Ezek segítségével egy lekérdezésből az eredetivel ekvivalens lekérdezést készítünk, amelynek kiszámítása az esetek többségében kevesebb I/O műveletet igényel majd.
- A *q, q'* relációs algebrai lekérdezések (vagy tetszőleges lekérdezések) ekvivalensek, ha tetszőleges *I* előfordulás esetén $q(I) = q'(I)$ fennáll. Jelben: $q \equiv q'$.

Egy példa…

- A táblák legyenek:

  Film (cím, év, hossz)

  Szerepel (filmcím, év, színésznév)

- Ekkor a következő lekérdezés:

  $\Pi_{\text{cím}}(\sigma_{\text{cím=filmcím} \wedge \text{F.év=Sz.év} \wedge \text{színésznév='Edus'}} (F \times Sz))$ ekvivalens a

  $\Pi_{\text{cím}}(\sigma_{\text{cím=filmcím} \wedge \text{F.év=Sz.év}}(F \times (\sigma_{\text{színésznév='Edus'}} (Sz))))$ lekérdezéssel.

- Emellett az utóbbi valószínűleg gyorsabban végrehajtható.

Descartes-szorzat és összekapcsolások

- Asszociativitás:

  $(E_1 \, \Theta \, E_2) \, \Theta \, E_3 \equiv E_1 \, \Theta \, (E_2 \, \Theta \, E_3)$, ahol $\Theta \in \{\times, |\triangleright\triangleleft|\}$ és

[természetes összekapcsolás]

  $(E_1 \; | \triangleright\triangleleft |_{F1} \; E_2) \; | \triangleright\triangleleft |_{F2} \; E_3 \equiv E_1 \; | \triangleright\triangleleft |_{F1} \; (E_2 \; | \triangleright\triangleleft |_{F2} \; E_3)$, ha

  $attr(F1) \subseteq attr(E1) \cup attr(E2)$ és $attr(F2) \subseteq attr(E2) \cup attr(E3)$

- [θ összekapcsolás]

- Kommutativitás:

  $E_1 \, \Theta \, E_2 \equiv E_2 \, \Theta \, E_1$, ahol $\Theta \in \{\times, | \triangleright\triangleleft |, | \triangleright\triangleleft |_F\}$.

Projekció és szelekció

- Projekció sorozat:

  $$\Pi_X(\Pi_Y(E)) \equiv \Pi_X(E), \text{ ha } X \subseteq Y.$$

- Kiválasztás és a feltételek konjunkciója:

  $$\sigma_{F1 \wedge F2}(E) \equiv \sigma_{F1}(\sigma_{F2}(E)).$$

- Kiválasztás és a feltételek diszjunkciója:

  $$\sigma_{F1 \vee F2}(E) \equiv \sigma_{F1}(E) \cup \sigma_{F2}(E).$$

- Kiválasztás elé projekció beillesztése:

  $$\Pi_X(\sigma_F(E)) \equiv \Pi_X(\sigma_F(\Pi_Y(E))), \text{ ahol } Y = \text{attr}(F) \cup X.$$

Kiválasztás és Descartes-szorzat/összekapcsolás

- Kiválasztás és Descartes-szorzat, összekapcsolás felcserélése:

$\sigma_F (E_1 \, \Theta \, E_2) \equiv \sigma_F (E_1) \, \Theta \, E_2$, ahol attr $(F) \subseteq$ attr (E_1) és $\Theta \in \{\times, \mid \triangleright\triangleleft \mid\}$.

- Általánosabban:

$\sigma_F (E_1 \, \Theta \, E_2) \equiv \sigma_{F1} (E_1) \, \Theta \, \sigma_{F2} (E_2)$, ahol attr $(F_i) \subseteq$ attr (E_i) $(i = (1, 2))$
$F = F_1 \wedge F_2$ és $\Theta \in \{\times, \mid \triangleright\triangleleft \mid\}$.

- Ezekből levezethető:

$\sigma_F (E_1 \, \Theta \, E_2) \equiv \sigma_{F2} (\sigma_{F1} (E_1) \, \Theta \, E_2)$, ahol attr $(F_1) \subseteq$ attr (E_1), $F = F_1 \wedge F_2$, de attr $(F_2) \subseteq$ attr (E_i) nem teljesül $(i = (1, 2))$, $\Theta \in \{\times, \mid \triangleright\triangleleft \mid\}$.

Projekció és Descartes-szorzat/összekapcsolás

- Projekció és Descartes-szorzat, összekapcsolás felcserélése:

$\Pi_X(E_1 \ \Theta \ E_2) \equiv \Pi_Y(E_1) \ \Theta \ \Pi_Z(E_2)$,

ahol $X = Y \cup Z$, $Y \subseteq attr(E_1)$, $Z \subseteq attr(E_2)$ és $\Theta \in \{\times, |\triangleright\triangleleft|\}$.

Projekció/kiválasztás és halmazműveletek

- Kiválasztás és unió (különbség) felcserélése:

  $\sigma_F (E_1 \Theta E_2) \equiv \sigma_F (E_1) \Theta \sigma_F (E_2)$, ahol $\Theta \in \{\cup, -\}$.

- Projekció unióval való felcserélése:

  $\Pi_X(E_1 \cup E_2) \equiv \Pi_X(E_1) \cup \Pi_X(E_2)$.

- Megjegyzés: nincs általános szabály a projekció különbséggel való felcserélésére.
- Kérdés: a metszettel mi a helyzet? [reláció séma]

Példa optimalizálásra

- A következő két feladathoz használt táblák:

  Személy (név, kor, város, ISBN)

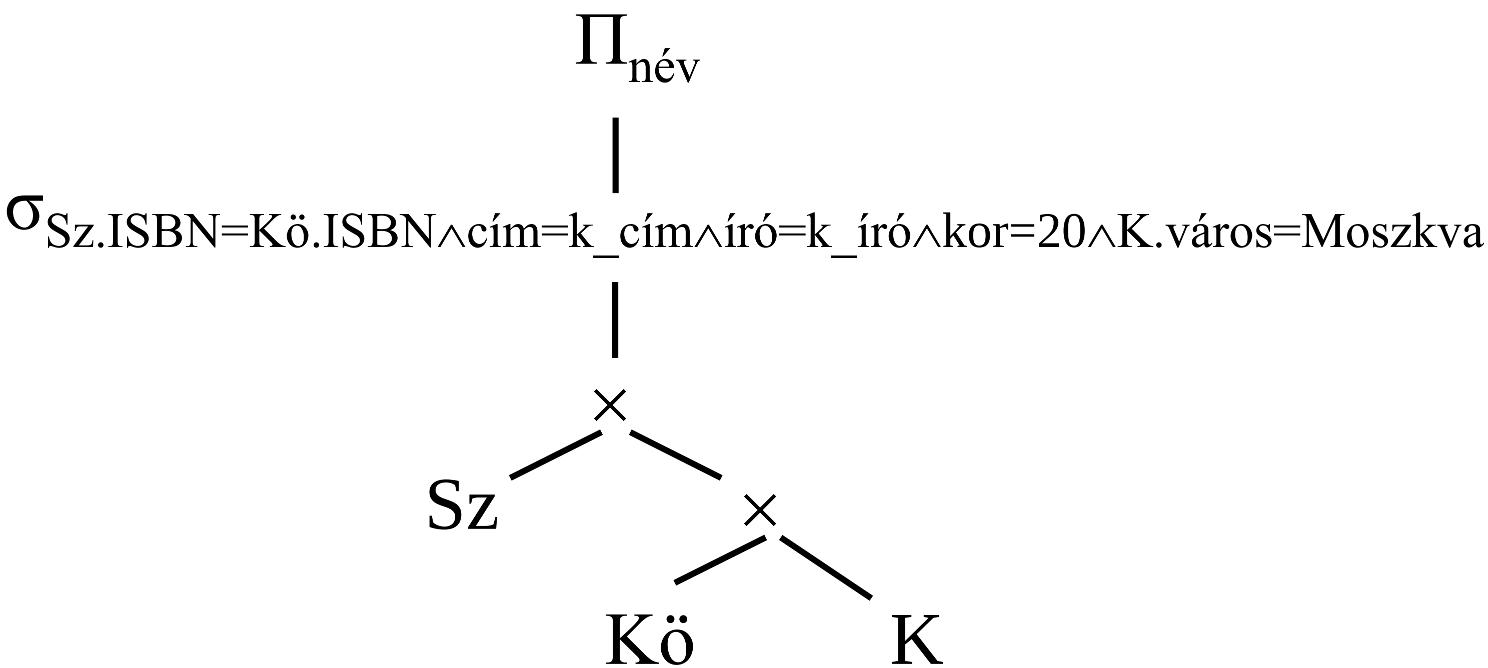
  Könyv (cím, író, ISBN, ár)

  Kiad (k_cím, k_író, város, ország)

- Kik azok, akik 20 évesek, és moszkvai kiadású könyvet kölcsönöztek ki?

$$\Pi_N(\sigma_{Sz.ISBN=Kö.ISBN \wedge cím=k\_cím \wedge író=k\_író \wedge kor=20 \wedge K.város=Moszkva}(Sz \times Kö \times K))$$

Lekérdezésfa



Kiválasztások "lejjebb csúsztatása"

- Első lépésben a kiválasztások konjunkciós feltételeit daraboljuk szét elemi feltételekké a $\sigma_{F1 \wedge F2}(E) \equiv \sigma_{F1}(\sigma_{F2}(E))$ szabály segítségével.
- Ezek után alkalmazzuk a kiválasztás halmazműveletekkel illetve Descartes-szorzattal és a természetes összekapcsolással való felcserélésének szabályait.
- Azaz: igyekszünk a kiválasztásokat minél hamarabb végrehajtani, hiszen azok jelentősen csökkenthetik a feldolgozandó köztes relációk méretét.
- A Théta-összekapcsolást itt jobb, ha egy Descartes-szorzatra és egy azt követő kiválasztásra bontjuk.

  $R \mid\triangleright\triangleleft\mid_F S \equiv \sigma_F(R \times S)$.

Darabolás

$$\Pi_{név}$$

$$\sigma_{Sz.ISBN=Kö.ISBN}$$

$$\sigma_{cím=k\_cím \wedge író=k\_író}$$

$$\sigma_{kor=20}$$

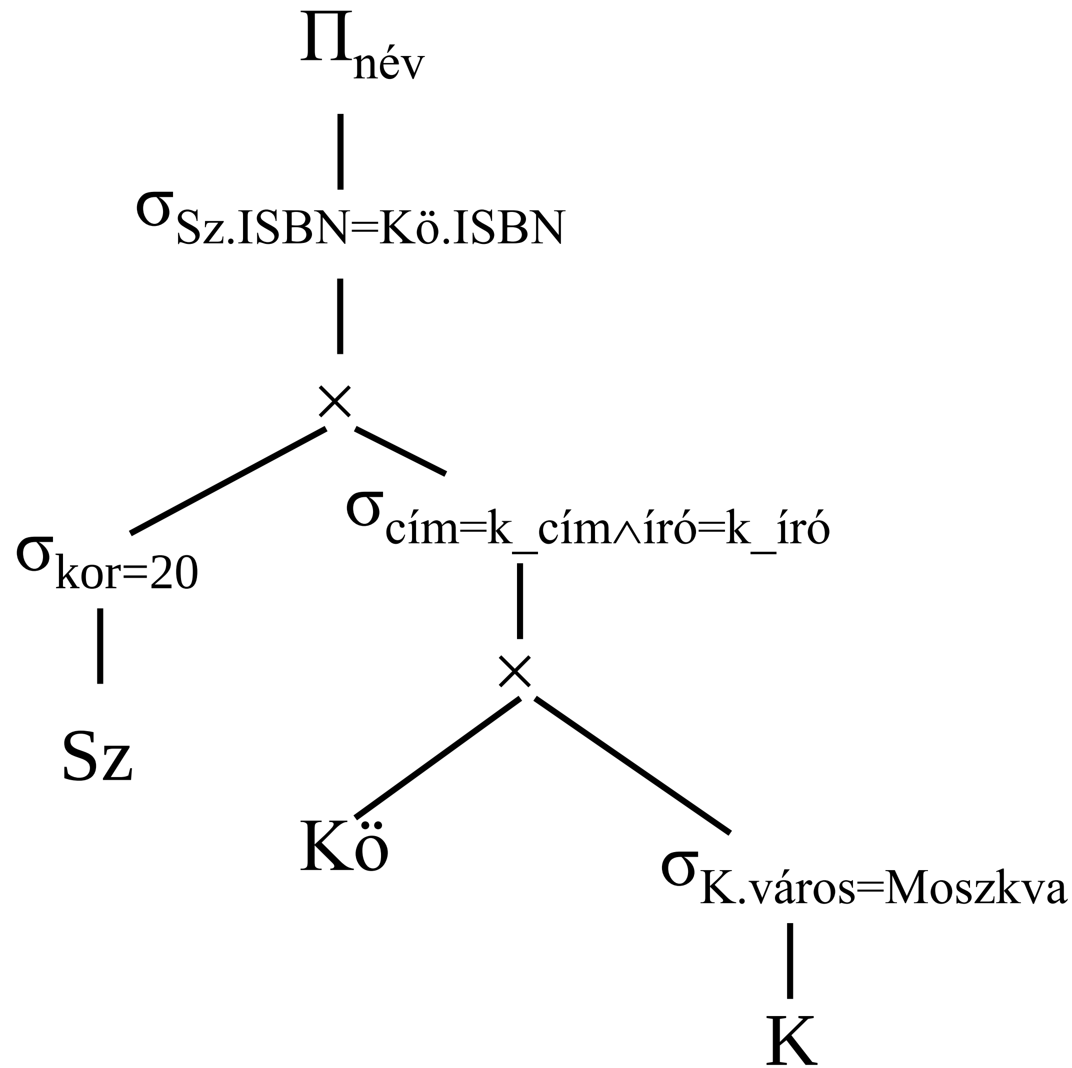
$$\sigma_{K.város=Moszkva}$$

$$\times$$

$$Sz \qquad \times$$

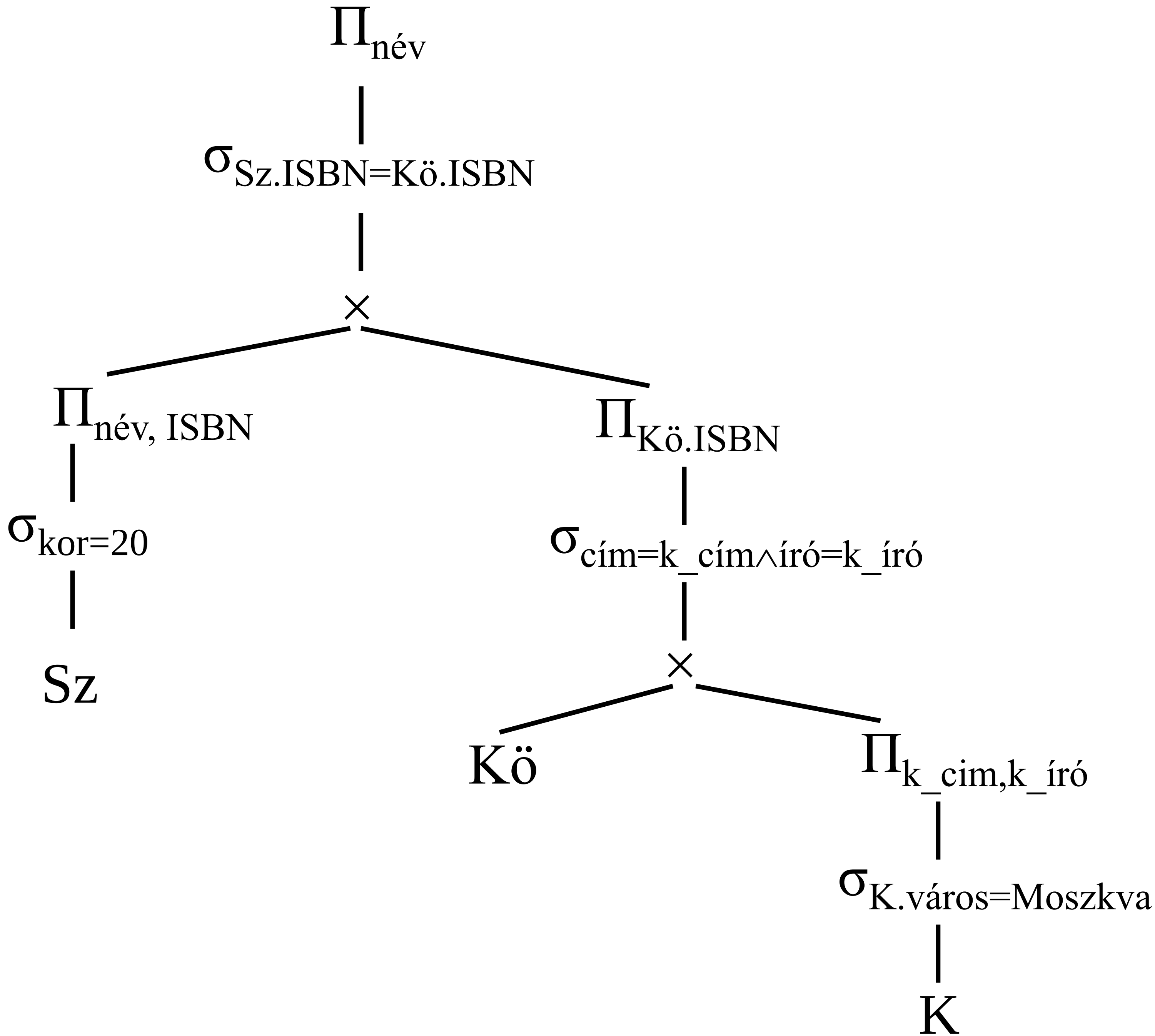
$$Kö \qquad K$$

Letolás


$\Pi_{\text{név}}$
$\sigma_{\text{Sz.ISBN=Kö.ISBN}}$
$\times$
$\sigma_{\text{kor=20}}$
$\sigma_{\text{cím=k\_cím} \wedge \text{író=k\_író}}$
Sz
$\times$
Kö
$\sigma_{\text{K.város=Moszkva}}$
K

Projekciók "beírása"

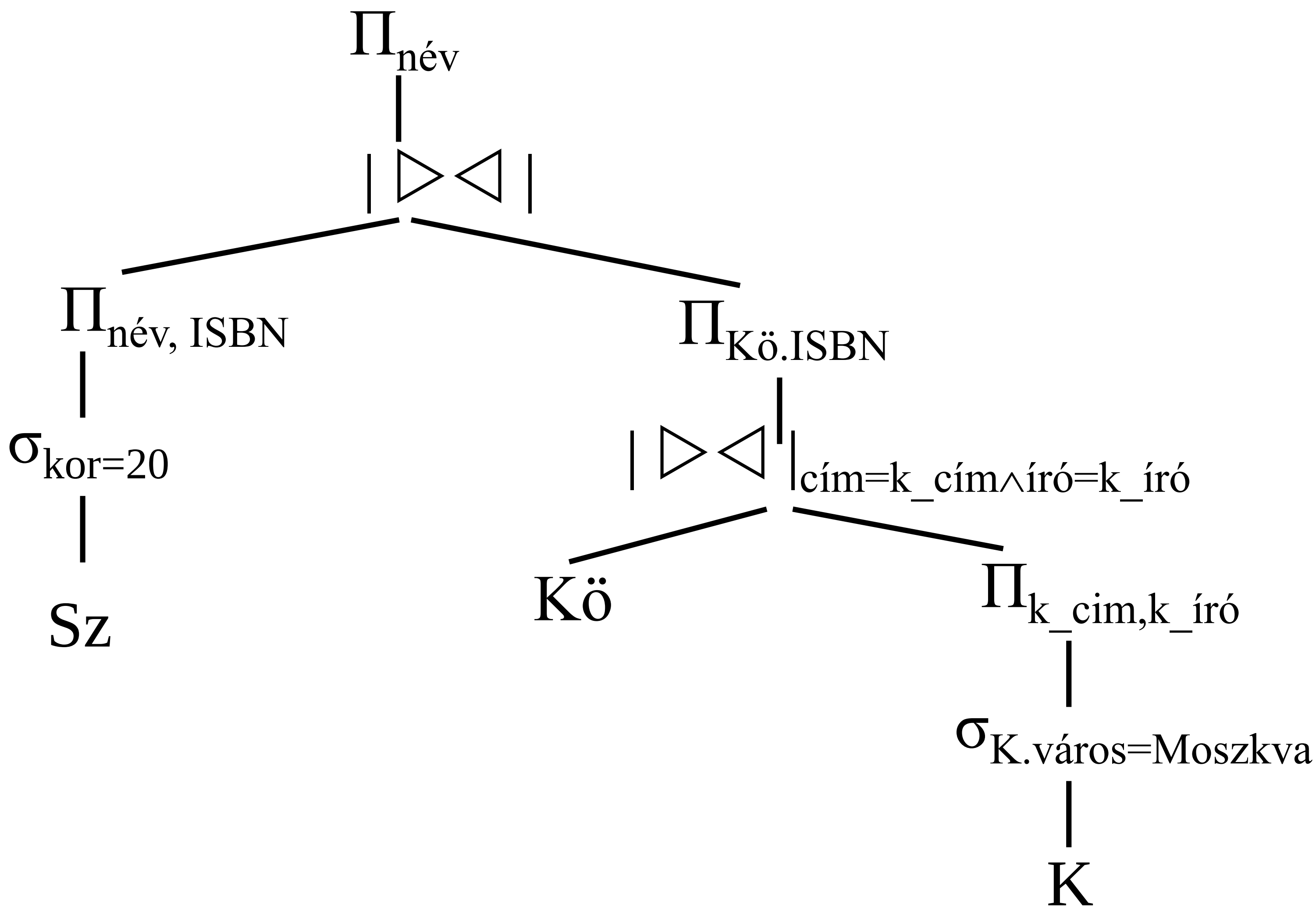
- Ennél a lépésnél igyekszünk csak azokat az oszlopokat megtartani a (köztes) relációkban, amelyekre később szükség lesz.
- Általában itt nem olyan nagy a nyereség. A projekciók végrehajtása viszont időt igényel, ezért meg kell gondolni, hogy tényleg végre akarjuk-e hajtani a vetítést.
- Az átalakításoknál értelemszerűen a projekciókra vonatkozó szabályokat használjuk.



$\Pi_{\text{név}}$
$\sigma_{\text{Sz.ISBN=Kö.ISBN}}$
$\times$
$\Pi_{\text{név, ISBN}}$
$\Pi_{\text{Kö.ISBN}}$
$\sigma_{\text{kor=20}}$
$\sigma_{\text{cím=k\_cím} \wedge \text{író=k\_író}}$
Sz
$\times$
Kö
$\Pi_{\text{k\_cim,k\_író}}$
$\sigma_{\text{K.város=Moszkva}}$
K

Összekapcsolások

- Az utolsó lépésben $\Pi_L(\sigma_C(R \times S))$, $\sigma_C(R \times S)$ kifejezéseket helyettesítjük természetes összekapcsolással, Théta-összekapcsolással úgy, hogy az eddigivel ekvivalens lekérdezést kapjunk.



Πnév
|▷◁|
Πnév, ISBN
σkor=20
Sz
ΠKö.ISBN
|▷◁|cím=k_cím∧író=k_író
Kö
Πk_cim,k_író
σK.város=Moszkva
K

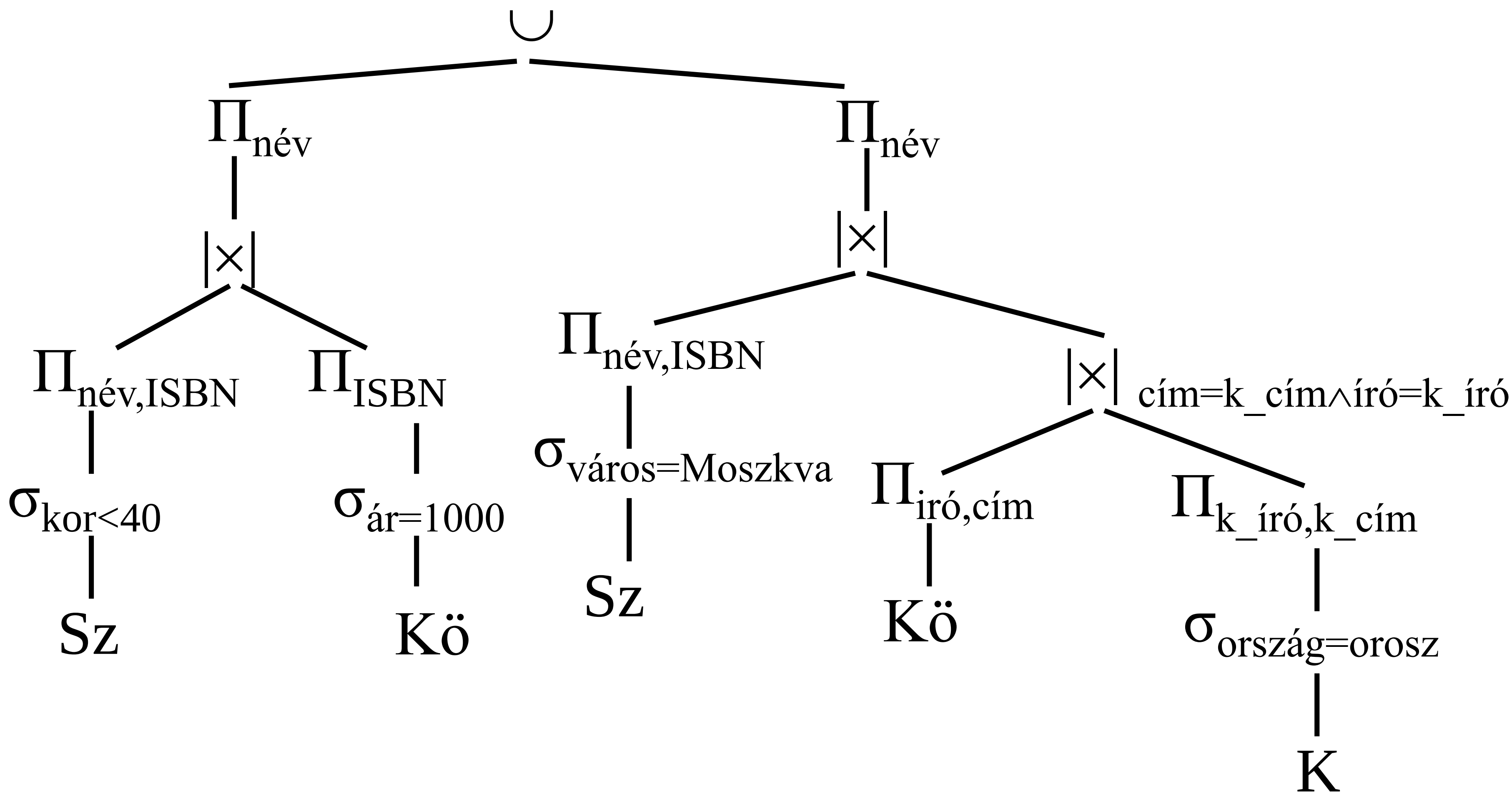
Mi történik, ha a diszjunkció is megjelenik?

- Kik azok, akik 1000 forintos könyvet vásároltak, és még nincsenek 40 évesek, vagy moszkvaiak, és orosz kiadású könyvet vettek?

$$\Pi_N(\sigma_{C\wedge((\text{ár}=1000\wedge \text{kor}<40)\vee(\text{Sz.város}=\text{Moszkva}\wedge \text{ország}=\text{orosz}))}(Sz \times Kö \times K)).$$

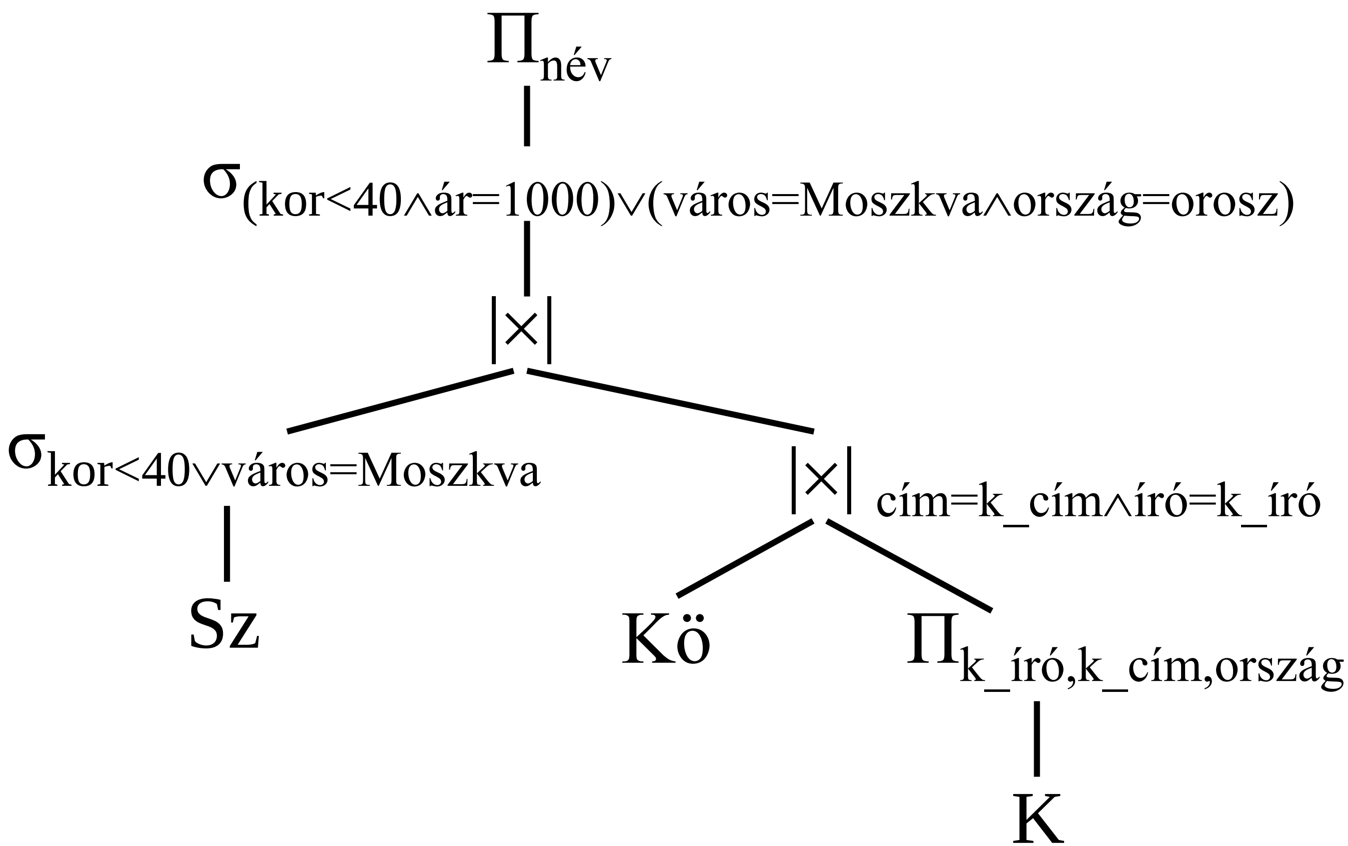
- Itt C az Sz.ISBN = Kö.ISBN ∧ Kö.cím = K.k_cím ∧ Kö.író = K.k_író feltételt jelöli.

Megoldás I.



∪
Π név
|×|
Π név,ISBN
σ kor<40
Sz
Π ISBN
σ ár=1000
Kö
Π név
|×|
Π név,ISBN
σ város=Moszkva
Sz
|×| cím=k_cím∧író=k_író
Π iró,cím
Kö
Π k_író,k_cím
σ ország=orosz
K

Megoldás II.


Π név
σ (kor<40∧ár=1000)∨(város=Moszkva∧ország=orosz)
|×|
σ kor<40∨város=Moszkva
|×| cím=k_cím∧író=k_író
Sz
Kö
Π k_író,k_cím,ország
K

Összegzés

- Ha tehát a kiválasztások feltételei diszjunkciót is tartalmaznak, a helyzet bonyolultabbá válik, és nem adható olyan egyértelmű optimalizációs algoritmus, mint konjunkciók esetén.

Kiválasztások feljebb csúsztatása

- A következő példa azt szemlélteti, amikor egy kiválasztást először felfelé kell csúsztatnunk, hogy aztán letolhassuk.
- A táblák:

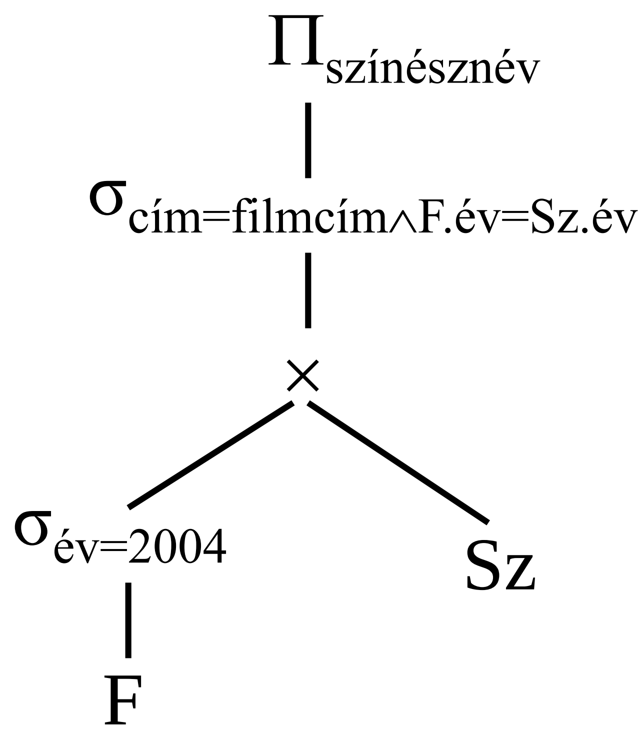
  Film (cím, év, hossz)

  Szerepel (filmcím, év, színésznév)

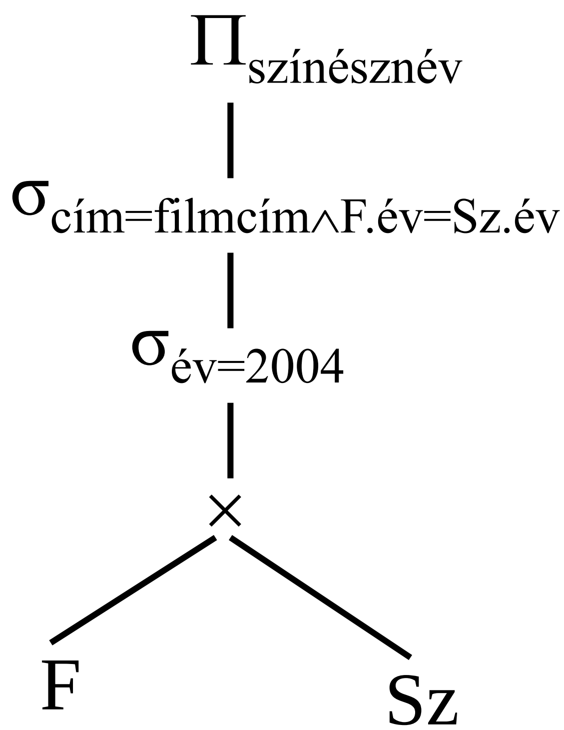
```
CREATE VIEW film04 AS
(SELECT *
FROM film
WHERE év = 2004);
```

```
SELECT színésznév
FROM film04 f, Szerepel sz
WHERE cím = filmcím AND
      f.év = sz.év;
```

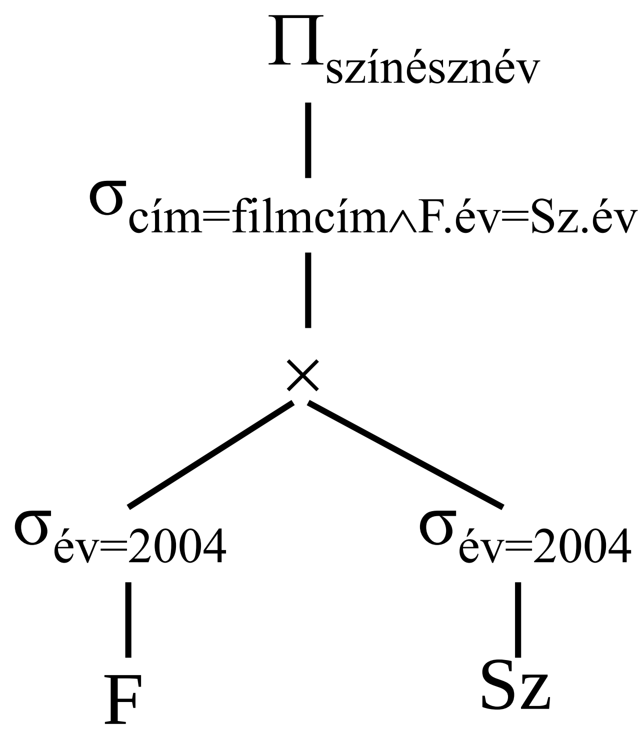
Kezdeti lekérdezésfa



Második lépés



És az eredmény…



Feladat

- A táblák legyenek:

  Film (cím, év, hossz)

  Szerepel (filmcím, év, színésznév)

  Színész (név, kor, város)

- Adjuk meg, hogy a nem budapesti, negyven évesnél idősebb színészek milyen filmekben játszottak 1998-ban. A lekérdezést optimalizáljuk.