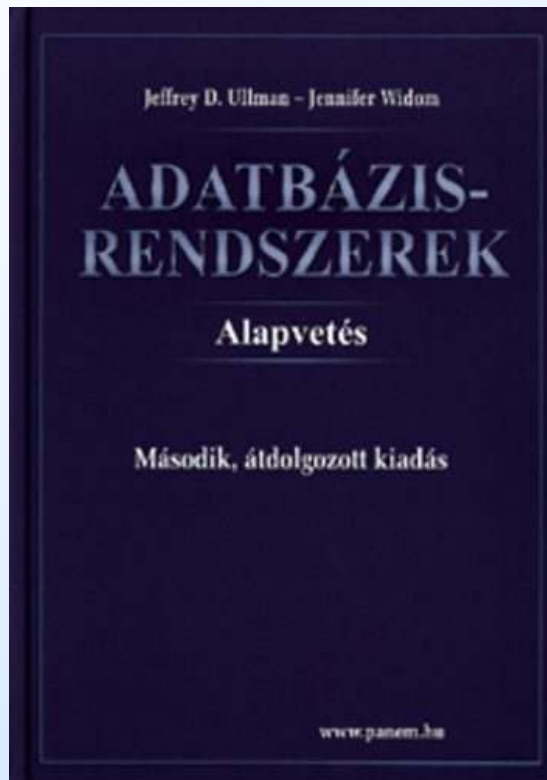


Rekurzív lekérdezések (az ELJUT feladat)



Ullman-Widom: Adatbázisrendszerek
Alapvetés. Második, átdolgozott kiadás,
Panem, 2009

10.2. Rekurzió Datalogban valamint
SQL-99-ben és SQL/PSM-ben

(Jeffrey D. Ullman, 2007 EA alapján)

Ismétlés: Rekurzió a gyakorlatban

- ◆ SQL-99 szabvány csak az ún. „monoton” rekurziót támogatja (viszont Datalogban megengedett a negáció és rekurzió együtt, de igen bonyolulttá válik a lekérdezés értelmezése, mint például a rétegzés),
- ◆ SQL-99 szabvány lineáris rekurziót enged meg, vagyis az értékadás kifejezésben egyetlen rekurzív relációt használhatunk.

Rekurzív Datalog

Tankönyv 10.2. fejezet példája (az ELJUT feladat)

- ◆ Jaratok(legitarsasag, honnan, hova, koltseg, indulas, erkezes) táblában repülőjáratok adatait tároljuk.

Mely (x,y) párokra lehet eljutni x városból y városba?

- ◆ Datalogban felírva (lineáris rekurzióval)

```
Eljut(x, y) <- Jaratok(_, x, y, _, _, _)
```

```
Eljut(x, y) <- Eljut(x, z) AND Jaratok(_, z, y, _, _, _)
```

- ◆ Datalogban felírva (nem-lineáris rekurzióval)

```
Eljut(x, y) <- Jaratok(_, x, y, _, _, _)
```

```
Eljut(x, y) <- Eljut(x, z) AND Eljut(z, y)
```

Rekurzív SQL

- ◆ Lineáris rekurzió kifejezhető az SQL-2003 szabványban

```
WITH RECURSIVE Eljut AS
```

```
(SELECT honnan, hova FROM Jaratok
```

```
UNION
```

```
SELECT Eljut.honnan, Jaratok.hova
```

```
FROM Eljut, Jaratok
```

```
WHERE Eljut.hova = Jaratok.honnan)
```

```
SELECT hova FROM Eljut WHERE honnan='Bp';
```

- ◆ Mivel az Oracle SQL a WITH RECURSIVE utasítást nem támogatja, ott másképpen kell megoldani!
- ◆ Relációs algebrában egy relációs algebrai kifejezéssel nem tudjuk előállítani, csak ciklusba szervezve...

Rekurzív Eljut feladat PSM-ben --1

- ◆ Az ELJUT feladatot a gyakorlaton oldjuk meg Oracle PL/SQL-ben, itt csak a vázlata PSM-ben
- ◆ A ciklus során ellenőrizni kell, hogy addig hajtsuk végre a ciklust, amíg növekszik az eredmény (Számláló)
- ◆ Eljut relációnak (nézettáblának) kezdeti értéket adunk (PL/SQL megvalósításnál az INSERT-nél figyelni, hogy ne legyenek ismétlődő sorok)
insert into eljut (SELECT distinct honnan, hova
FROM jaratok);

Rekurzív Eljut feladat PSM-ben --2

- ◆ Szamlalo változónak adunk értéket:

```
RegiSzamlalo:=0;
```

```
select count(*) into UjSzamlalo from eljut;
```

- ◆ A ciklust addig kell végrehajtani, amíg növekszik az eredmény (Szamlalo) duplikátumokra figyelni!

LOOP

```
insert into eljut (lásd a köv.oldalon...)
```

```
select count(*) into UjSzamlalo from eljut;
```

```
EXIT WHEN UjSzamlalo = RegiSzamlalo;
```

```
RegiSzamlalo := UjSzamlalo;
```

```
END LOOP;
```

Rekurzív Eljut feladat PSM-ben --3

- ◆ Az eljut tábla növelése a ciklusban, figyelni kell a duplikátumokra, csak olyan várospárokat vegyünk az eredményhez, ami még nem volt!

insert into eljut1

```
(select distinct eljut1.honnan, jaratok.hova  
from eljut1, jaratok  
where eljut1.hova = jaratok.honnan  
and (eljut1.honnan,jaratok.hova)  
NOT IN (select * from eljut1));
```