

Rekurzív lekérdezések: Az „Eljut-feladat”

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

10.2. Rekurzív lekérdezések a Datalogban valamint az SQL-99 szabványban

-- intiutív bevezetés a Datalogba:

-- Az „Eljut”-feladat („Jaratok”- táblából honnan-hova lehet eljutni el?)



Az „Eljut-feladat” Datalogban

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** (intuitív bevezetés a Datalogba)

$\text{Eljut}(x, y) \leftarrow \text{Jaratok}(l, x, y, k, i, e)$

$\text{Eljut}(x, y) \leftarrow \text{Eljut}(x, z) \text{ AND } \text{Jaratok}(l, z, y, k, i, e)$

- Vagy **másképp felírva** Datalogban (mi a különbség?)

$\text{Eljut}(x, y) \leftarrow \text{Jaratok}(_, x, y, _, _, _)$

$\text{Eljut}(x, y) \leftarrow \text{Eljut}(x, z) \text{ AND } \text{Eljut}(z, y)$

Az „Eljut feladat” SQL-99 szabványban

- Datalog **lineáris és nem-monoton rekurzió** átírható:
Eljut(x, y) <- Jaratok(l, x, y, k, i, e)
Eljut(x, y) <- Eljut(x, z) AND Jaratok(l, z, y, k, i, e)

- Hova, mely városokba tudunk eljutni Budapestről?

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';

Eljut feladat relációs algebrában (nincs mo)

- Az **Oracle SQL** a WITH RECURSIVE utasítást nem támogatja, ott másképpen oldották meg WITH utasítással (Oracle 11gR2 verziótól), illetve Oracle-ben már jóval korábban volt hierarchikus lekérdezés CONNECT BY záradékkal (lásd a gyakorlati anyagban)
- **Relációs algebrában** egy relációs algebrai kifejezéssel nem tudjuk előállítani, csak ciklusba szervezve... nézzük meg PSM-ben:

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)
- **DECLARE** RegiSzamlalo Integer;
UjSzamlalo Integer;
- Deklarációs rész után **BEGIN ... END;** között az utasítások, először az eljut táblának kezdeti értéket adunk (a megvalósításnál az INSERT-nél figyelni, hogy ne legyenek ismétlődő sorok: select distinct)
delete from eljut;
insert into eljut (**SELECT distinct** honnan, hova
FROM jaratok);

Rekurzív Eljut feladat PSM-ben ---2

- Szamlalo változóknak adunk kiindulási értéket:
SET 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;  
SET 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 eljut

```
(select distinct eljut.honnan, jaratok.hova  
from eljut, jaratok --- *from (lineáris rekurzió)  
where eljut.hova = jaratok.honnan  
and (eljut.honnan,jaratok.hova)  
NOT IN (select * from eljut));
```

- Megjegyzés: PSM-ben a **nem-lineáris rekurzió** is megengedett: **from eljut e1, eljut e2 ---*from-ban**

Milyen fontos rekurzív feladatok vannak?

I. Hierarchiák bejárása

- **Leszármazottak-ősök** ParentOf(parent,child)
 - Find all of Mary's ancestors
- **Vállalati hierarchia felettes-beosztott**
Employee(ID,salary)
Manager(mID,eID)
Project(name,mgrID)
 - Find total salary cost of project 'X'
- **Alkatrész struktúra** (mely alkatrésznek mely alkatrész része)

Milyen fontos rekurzív feladatok vannak?

II. Gráf jellegű bejárások

■ Repülőgép járatok, eljut-feladat

Flight(orig,dest,airline,cost)

- Find cheapest way to fly from 'A' to 'B'

■ Közösségi hálók

Ki-kinek az ismerőse, Twitterben ki-kit követ

Kiegészítés a gráf adatbázisokról

- Gráfok könnyen megadhatók relációs táblával, a gráf lekérdezések egyre gyakoribb feladatok, ezek relációs megoldása hatékonysági kérdés. Vannak kimondottan gráf-adatbázisok.

Következik...

- Nézzünk meg olyan logikai felépítést, amivel az ilyen rekurzív jellegű lekérdezések könnyen megoldhatók.
- Ez a nyelv: a Datalog (Tankönyv 5.3-5.4)