9. tétel
A *Q(A,B) JOIN R(B,C) JOIN S(C,D)* háromféle kiszámítási módja és költsége, (feltéve, hogy Q,R,S paraméterei megegyeznek, Q.B-re és S.C-re klaszterindexünk van).
a) balról jobbra,
b) balról jobbra és a memóriában összekapcsolva a harmadik táblával,
c) a középső ténytábla soraihoz kapcsolva a szélső dimenziótáblákat.

Feltevések:
$T_Q = T_R = T_S = T$ (ugyanannyi soruk van)
$B_Q = B_R = B_S = B$ (ugyanannyi helyet foglalnak)
$I_{Q.B} = I_{R.B} = I_{R.C} = I_{S.C} = I$ (a képméretek, vagyis az előforduló értékek száma azonos)

Most az előállított eredményt ki is írjuk a lemezre, vagyis az output kiírásának költségét is belefoglaljuk a költségbe mindhárom esetben.

Az ***a)* esetben** összekapcsoljuk *Q* és *R* táblákat egy index segítségével, az eredményt kiírjuk egy átmeneti *TMP* táblába (**materializáció**), majd a *TMP* táblát kapcsoljuk össze *S*-el az *S*-en levő index segítségével.
A ***b)* esetben** csupán annyi lesz a változás, hogy nem írjuk ki a *TMP* táblát a lemezre, hanem az előállítása közben rögtön elvégezzük az indexes összekapcsolást az *S* táblával (**futószalagosítás**).
A ***c)* esetben** az *R* táblát olvassuk be először, majd a sorokhoz az indexek segítségével olvassuk be *Q* és *S* sorait és rögtön el is végezzük a sorok összekapcsolását.

Előzetes számítások és képletek
Az alábbi képletek már mind szerepeltek a korábbi előadásokon, de ismétlésképpen leírjuk őket még egyszer, mert fel fogjuk használni a későbbiekben.
Először nézzük meg, hogyan lehetne előállítani két tábla összekapcsolását *R(A,B) JOIN S(B,C)*-t, ha az egyik táblán (S-en) van index a közös oszlopra. Az első tábla sorait beolvassuk a memóriába, a második táblából az azonos értékekhez tartozó sorokat az index alapján olvassuk be, majd a memóriában összekapcsoljuk a sorokat. Feltesszük, hogy az összekapcsolandó sorok beférnek a memóriába (vagyis $B_R/I + B_S/I <= M$, és így nem kell újra beolvasnunk egyetlen sort sem), továbbá, hogy az indexek kis méretűek és így végig a memóriában tartjuk őket, valamint, hogy R.B részhalmaza S.B-nek. Egy index segítségével történő beolvasás költsége, így megegyezik a beolvasott blokkok számával, vagyis $B_S/I_{R.S}$-sel.

Az indexes **JOIN művelet I/O költsége** (beolvassuk R-et, majd minden sorához index segítségével S-et. [optimization.ppt 30. old.] Az alábbi képlet az output kiírásának költségét nem tartalmazza.)
$B_R + T_R * B_S/I_{S.B}$ $I_{R.B}=I_{S.B} = I$ esetén:

(1) $\mathbf{B_R + T_R*B_S/I}$

Hány sora lesz a JOIN-nak?
$T_{R|><|S} = I_{R.B} * (T_R/I_{R.B} * T_S/I_{S.B})$ (optimization.ppt 35. old., output_estimate.ppt 16. old.))

Ha feltesszük, hogy $I_{R.B}=I_{S.B} = I$, akkor **a JOIN sorainak száma:**

(2) $\mathbf{T_{R|><|S} = T_R*T_S/I}$

Mekkora méretű lesz az output? (RxS esetén $T_R * B_S + T_S * B_R$ lenne: fizika.ppt 3. old.)
Az output mérete:

(3) $(T_R*B_S + T_S*B_R)/I$

A fenti 3 képletet fogjuk felhasználni a *Q(A,B) JOIN R(B,C) JOIN S(C,D)* kiszámításához.

a) balról jobbra történő kiszámítás
Q(A,B) JOIN R(B,C)-re
Output mérete: B_{TMP} = **2*T*B/I** lásd (3)
Sorok száma: T_{TMP} = T^2/I lásd (2)
I/O költség: **B + T*B/I** lásd (1)

Használjuk fel a fentieket *Q(A,B) JOIN R(B,C) JOIN S(C,D)* esetén az output és az I/O költség kiszámításához.
Output mérete (3)-ba helyettesítve:

$$(T_{TMP}*B_S + T_S*B_{TMP})/I \ \text{->}\ [(T^2/I)*B + T*(2*T*B/I)]/I = 3*T^2*B/I^2$$

A teljes JOIN I/O költsége:
Az 1. join költsége **B + T*B/I**
Az 1. join kiírása (output mérete): **2*T*B/I**
A 2. join költsége ($B_{TMP} + T_{TMP} * B/I$): $2*T*B/I + [(T^2/I)*B]/I$
A teljes output kiírása: $3*T^2*B/I^2$
összesen:

a) végeredménye: $B + 5*T*B/I + 4*T^2*B/I^2$

b) balról jobbra és a memóriában összekapcsolva a harmadik táblával,
Megspórolhatjuk az 1. join eredményének kiírását majd újbóli beolvasását, vagyis
2* (2*T*B/I)-t. Az eredmény ekkor:

b) végeredménye: $B + T*B/I + 4*T^2*B/I^2$

c) a középső ténytábla soraihoz kapcsolva a szélső dimenziótáblákat.
Beolvassuk R-et, majd R minden sorára index alapján olvassuk be Q és S sorait. A költség ekkor:
R beolvasása **B**
Q és S olvasása R minden sorára: **T*(B/I + B/I)**
A teljes output kiírása: $3*T^2*B/I^2$
összesen:

c) végeredménye: $B + 2*T*B/I + 3*T^2*B/I^2$

Nézzük meg, hogy a b) és c) esetek közül melyik a kisebb költségű. A két költség közötti különbség (b-c): $T^2*B/I^2 - T*B/I$

Nagyméretű táblák esetén a T/I hányados nagy szám lesz, ezért a négyzetes tag jóval nagyobb lesz, mint a lineáris tag, vagyis a c) módszer a leghatékonyabb.

Ha a c/b arányt tekintjük, akkor azt mondhatjuk, hogy ez az arány ¾-hez tart, ha T/I tart a végtelenbe. Vagyis ha T/I elég nagy, akkor a c költsége nagyjából ¾-e a b-nek.