DB2 and DBSA – April 11, 2017 – First Intermediate Test

You can answer this test in English, Italian, or any mixture

1. Consider a schema R(IdR, A, B, …, IdS*), S(IdS,…), T(IdT, …, IdS*) and the following query

```
SELECT   *
FROM     R, T
WHERE    R.IdS = T.IdS And R.A ≤ 10 And R.B ≤ 200
```

Assume that R, S and T are stored as heap files. Primary keys are R.IdR, S.IdS and T.IdT, while R.IdS abd T.IdS are foreign keys that refers to S.
Assume that unclustered RID-sorted index are defined on all the primary and foreign keys, on R.A, on R.B, and a combined index on (R.A,R.B). Assume that every index on R has 4.000 leaves, every index on S has 10.000 leaves, every index on T has 200 leaves **(some part of this information is not needed for this exercise)**.
Assume the following table for the optimization parameters. If you need Cardenas formula Φ(n,k), approximate it with min(n,k).
Assume that every page is 4.000 bytes long, and that every attribute uses 4 bytes.

	NReg	NPag	NLeaf of Indexes	NKey	Min	Max
R	2.000.000	100.000	4.000			
S	5.000.000	500.000	10.000			
T	100.000	1.000	200			
Idx.R.A			See R	100	0	1.000
Idx.R.B			See R	10	0	1.000

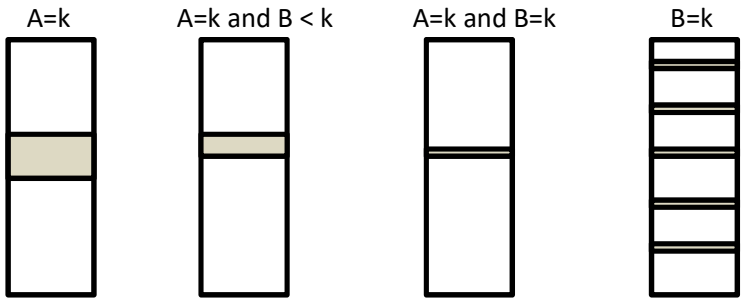
   a) Compute the cost of accessing all records of R that satisfy the condition R.A ≤ 10 And R.B ≤ 200 using:
      a. No index
      b. Index on R.A only
      c. Index on R.B only
      d. Both R.A and R.B indexes
   b) Using the cheapest access plan from (a), compute the cost of an IndexNestedLoop plan for the complete query, where R is the outer relation
   c) Compute the cost of an IndexNestedLoop plan for the complete query, where R is the inner relation
   d) How many records are in the result of the query?

2. Consider a combined index on attributes (A,B) of a table T. Assume the standard hypothesis of uniformity and independence of T.A and T.B, and also that, for every a in $\pi_A(T)$ and b in $\pi_B(T)$, the pair (a,b) belongs to $\pi_{AB}(T)$. The following table reports the cost of accessing the data if the index is unclustered with sorted RIDs, if it is clustered on (A,B), or if it is unclustered with unsorted RIDs. The table assumes that RIDS are used to fetch the record in the same order as they are found in the index, we do not merge different RID lists and we do not sort RIDs in main memory.

We use the following abbreviations for the involved selectivity factors – Ae stands for 'equality condition on A' and Ar 'range condition on A'.
sfAe = 1/Nkey(A)
sfAr = (k – min(A)) / (max(A)-min(A))
sfBe = 1/Nkey(B)
sfBr = (k' – min(B)) / (max(B)-min(B))
sfXxYy = sfXx * sfYy

Condition	Unclustered sorted RIDs	Clustered on A, B	Uncl. Unsorted RIDs
A=k	Nkey(B) *⌈Φ(sfAeBe*Nrec,Npag)⌉	sfAe*Npag	
A<k	⌈sfAr*NKey(A)*Nkey(B)⌉ *⌈Φ(sfAeBe*Nrec,Npag)⌉	sfAr*Npag	
B=k'	Nkey(A) *⌈Φ(sfAeBe*Nrec,Npag)⌉		
B<k'		Nkey(A)*⌈sfAeBr*NPag⌉	
A=k and B=k'	⌈Φ(sfAeBe*Nrec,Npag)⌉		
A=k and B<k'	⌈sfBr*NKey(B)⌉ *⌈Φ(sfAeBe*Nrec,Npag)⌉	sfAeBr*Npag	
A<k and B=k'			
A<k and B<k'	⌈sfArBr*NKey(A)*Nkey(B)⌉ *⌈Φ(sfAeBe*Nrec,Npag)⌉	sfAr*Nkey(A)*⌈sfAeBr*NPag⌉	sfArBr*Nrec

a. Consider the following graphical representation of the records that satisfy four of the conditions of the table below in a file that is sorted on (A,B). Complete that with the representation of the missing four conditions.



b. Draw a picture of an inverted-list combined index on (A,B)
c. Consider the case of 'A<k and B<k': explain how data is retrieved, in the three cases of the table, and also explain the three costs that are reported in the table
d. Give a general formula for the first and for the third columns – you may you sf(ϕ) for the selectivity factor of the condition in the current line
e. (Optional) Fill up the holes in second column

3. Consider the following log content. Assume that the DB was identical to the buffer before the beginning of this log, and consider a undo-redo protocol

(begin,T1) (W,T1,A,1,30) (begin,T2) (W,T2,B,1,20) (begin-ckp,{T1,T2}) (begin,T3) (W,T3,C,1,40) (commit,T1) (W,T2,A,30,50) (end-ckp) (begin,T4) (commit,T3) (W,T4,C,40,60)

a. Before starting this log, what was the content of A, B and C in the PS (Persistent Store)?
b. Assume there was a crash at the end of the logging period. At crash time, what was the content of A, B and C in the buffer? What can be said about the content of A, B and C in the PS?
c. At restart time, which transactions are undone? Which are redone?
d. List the operations that are redone, in the order in which are redone
e. After restart is finished, what is the content of A, B and C in the buffer?
f. Undo and Redo are executed in the buffer or on the PS?
g. After restart is finished, what is the content of A, B and C in the PS?
h. Assume now a Redo-NoUndo protocol. In this case, what can be said about the content of A, B and C in the PS at crash time, that is, after the completion of (W,T4,C,40,60)?

4. Assume that a system with no scheduler produces the following history, where we omit the commits:

$r_1[C], r_2[A], r_1[C], w_3[B], w_2[B], w_1[C], c_1, r_2[A], c_2, w_3[B], c_3$

a) Is this history serializable?
b) Exhibit a history that may be produced by a strict 2PL scheduler if presented with the above operations in that order, assuming that each transaction commits immediately after its last operation. In case of deadlock, assume that a transaction is aborted and is restarted later
c) Repeat (b) considering now a Wait-Die 2PL scheduler