Database Systems: 2nd Mid

Date: 15th March 2019 **Duration: 1.5 hrs**

1. **No clarifications during the exam.**
2. **Make *reasonable assumptions* and *clearly state* them to answer *ambiguous* questions.**
3. **Show your steps. Be concise and organized.**
4. **Calculators allowed. Sharing of calculators *not* allowed.**

1) If B(S) = B(R) = 10,000 and M=1000, what is the number of disk I/O's required for a (a) hash join, (b) hybrid hash join. Explain your calculations/formulae. [10]

2) Starting with an expression $\pi_L(R(a,b,c) \bowtie S(b,c,d,e))$ push the projection down as far as it can go if L is:
(a) $b+c \rightarrow x,\ c+d \rightarrow y$ (b) $a,\ b,\ a+d \rightarrow z$ [10]

3) (a) Estimate the size of the join $R(a,b) \bowtie S(b,c)$ using histograms for $R.b$ and $S.b$. Assume $V(R,b)=V(S,b)=20$, $T(R) = 52$, $T(S) = 78$, and the histogram for both attributes give the frequency of the four most common values, as tabulated below:

	0	1	2	3	4	others
R.b	5	6	4	5		32
S.b	10	8	5		7	48

(b) How would you estimate the join size if you did not have the above histograms? [10]

4) The following is a sequence of undo-log records written by two transactions T and U: <START T>; <T, A, 10>; <START U>; <U, B, 20>; <T, C, 30>; <U, D, 40>; <COMMIT U>; <T, E, 50>; <COMMIT T>. Describe the action of the recovery manager, including changes to both disk and the log, if there is a crash and the last log record to appear on disk is:

a. <START U>
b. <COMMIT U>
c. <T, E, 50>
d. <COMMIT T>

[10]

5) Consider the following sequence of undo/redo log records: <START S>; <S, A, 60, 61>; <COMMIT S>; <START T>; <T, A, 61, 62>; <START U>; <U, B, 20, 21>; <T, C, 30, 31>; <START V>; <U, D, 40, 41>; <V, F, 70, 71>; <COMMIT U>; <T, E, 50, 51>; <COMMIT T>; <V, B, 21, 22>; <COMMIT V>. Suppose that we begin a non-quiescent checkpoint immediately after one of the following log records has been written (in memory):

a. <S, A, 60, 61>
b. <T, A, 61, 62>
c. <T, E, 50, 51>

For each, tell:
i. At what points could the <END CKPT> record be written, and
ii. For each possible point at which a crash could occur, how far back in the log we must look to find all possible incomplete transactions. Consider both the case that the <END CKPT> record was or was not written prior to the crash.

[10]