Lab 8

Due: 04 Dec 2022, 11.59 PM

1. The lost update anomaly is said to occur if a transaction T_j reads a data item, then another transaction T_k writes the data item (possibly based on a previous read), after which T_j writes the data item. The update performed by T_k has been lost, since the update done by T_j ignored the value written by T_k . **[5+5 Marks]**

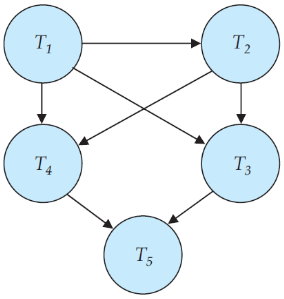

Fig: Precedence graph

a. Give an example of a schedule showing the lost update anomaly.

b. Give an example schedule to show that the lost update anomaly is possible with the **read committed** isolation level.

2. You must explain how you reach to an answer. Writing a number without explanation is not accepted.

[3+2+5 Marks]

- Consider a disk with block size B = 512 bytes.
- A block pointer is P = 6 bytes long, and a record pointer is RP = 7 bytes long.
- A file has r = 30,000 EMPLOYEE records of fixed length.
- Each record has the following fields: Name (30 bytes), SSN (9 bytes), Department_code (9 bytes), Address (40 bytes), Phone (10 bytes), Birth_date (8 bytes), Sex (1 byte), Job_code (4 bytes), and Salary (4 bytes, real number).
- An additional byte is used as a deletion marker.

a) Calculate the record size R in bytes. (A record is composed of fields. A field holds information about an entity. For example, Name and SSN are two fields of an employee's record)

b) Calculate the blocking factor *bfr* and the number of file blocks *b*, assuming an un-spanned organization.

c) Suppose that the file is ordered by the key field SSN and we want to construct a **primary index** on SSN. Calculate:

(i) the index blocking factor bfri (which is also the index fan-out fo);

(ii) the number of first-level index entries and the number of first-level index blocks;

(iii) the number of levels needed if we make it into a multilevel index;

(iv) the total number of blocks required by the multilevel index; and

(v) The number of block accesses needed to search for and retrieve a record from the file—given its SSN value—using the one-level primary index and multilevel index.

3. Consider the following relational schema. An employee can work in more than one department; the *pct time* field of the Works relation shows the percentage of time that a given employee works in a given department. **[20 Marks]**

Emp(*eid:* integer, *ename:* string, *age:* integer, *salary:* real)
Works(*eid:* integer, *did:* integer, *pct time:* integer)
Dept(*did:* integer, *budget:* real, *managerid:* integer)

Write SQL integrity constraints (domain, key, foreign key, or CHECK constraints) and SQL triggers to ensure each of the following requirements, considered independently.

1. Employees must make a minimum salary of $1000.

2. Every manager must be also be an employee.

3. The total percentage of all appointments for an employee must be under 100%.

4. A manager must always have a higher salary than any employee that he or she manages.

5. Whenever an employee is given a raise, the managers salary must be increased by the same percentage.