Washington State University

School of Electrical Engineering and Computer Science

CptS 451 – Introduction to Database Systems

Online

Dr. Sakire Arslan Ay

Homework-6

Name: _____

Student Number:_____

Question:	Max points:	Score:
1	100	
2 (extra credit)	15	
Total	100	

(100pts) Question1: Indexing

Consider the following relational schema for a portion of a university database:
Prof(ssno, pname, office, age, sex, specialty, dept_did)
Dept(did, dname, budget, num_majors, chair ssno)

Note that:
- ssno is the primary key for **Prof** and did is the primary key for **Dept**.
- **Prof**.dept_did is a foreign key referencing **Dept**.did
- Each professor is involved with some department.

Suppose you know that the following queries are the five most common queries in the workload for this university and all five are roughly equivalent in frequency and importance:

1. List the names, ages, and offices of professors of a user-specified sex (male or female) who have a user-specified research specialty (e.g., artificial intelligence). Assume that the university has a diverse set of faculty members, making it very uncommon for more than a few professors to have the same research specialty.
2. List all the department information for departments with professors in a user specified age range.
3. List the department id, department name, and chairperson name for departments with a user-specified number of majors.
4. List the lowest budget for a department in the university.
5. List all the information about professors who are department chairpersons.

These queries occur much more frequently than updates, so you should build whatever indexes you need to speed up these queries. However, you should not build any unnecessary indexes (or include any unnecessary attributes in an index), as updates will occur (and would be slowed down by unnecessary indexes). Given this information, decide which attributes should be indexed and whether each index should be a clustered index or an unclustered index. Assume that both B+ trees and hashed indexes are supported by the DBMS, and that both single- and multiple-attribute index keys are permitted.

For each index:
- identify the attributes you recommend indexing on,
- indicate whether each index should be clustered or unclustered, and
- whether it should be a B+ tree or a hashed index.

(Extra credit: 15 pts) Question 2: Storage and Indexing
Consider the relation below:
Student (sid, sname, major, email)

- The s*id* is a key (i.e., *sid* values are unique).
- Assume *sid* values are uniformly distributed between '100' and '204,900'
- All attributes have type char(40) (i.e., each attribute's size is 40 bytes).
- The relation contains 100,000 records (assume fixed length records)
- Block size is 16KB+8byte (assume each page has additional 8 bytes to store the pointer to next page)
- Assume the time to read/write to/from a page is D; assume the records are compacted and there is no gap between records.
- Assume each record pointer (RID) size is 8 bytes.
- Assume 1KB= 1000 bytes

Show your work for all questions below.

(a) **(10pts)** Assume relation Student is stored in a heap file. What is the cost of (i) file scan, (ii) equality search (sid='25700'), (iii) range search (sid<='25700') on Student?

(b) **(10pts)** Assume there is a clustered B+ tree index on *sid* using alternative-1 for relation Student. What is the cost of (i) file scan, (ii) equality search (sid='25700'), (iii) range search (sid<='25700') on Student? (assume the B+tree has 67% occupancy, i.e., the physical data pages are 1.5 times more than original data file; assume the height of the B+tree is 3.)

(c) **(10pts)** Assume there is an unclustered B+ tree index on *sid* using alternative-3 for relation Student. What is the cost of (i) file scan , (ii) equality search (sid='25700'), (iii) range search (sid<='25700') on Student? (assume the B+tree has 67% occupancy, i.e., the index pages are 1.5 times more than sequential index. Assume the height of the B+tree is 3.)

Additional notes:

1. To calculate selectivity for a range query, use the following formula:
   Assuming the values of attribute A is uniformly distributed between X1 and X2. The selectivity of a query with range condition `lower < A< upper` will be:
   `(upper - lower) / (X2-X1)`
   Note that if lower is skipped then you can use X1 instead of lower.
2. If the calculations result in flowt values, you may round the values up or truncate them as needed.

Submission Instructions:
Homework 6 will be submitted electronically on Blackboard to HW6-DROPBOX. You may either type/draw your HW in an editor or handwrite it and then scan it. Do not take pictures of your handwritten pages. Name your file CptS451_HW6_<yourname>.pdf Please submit only PDF files.