CS 34800 Homework 6

Deadline for returning the HW: Monday April 9th, 2012 at the end of the CS 34800 class.

Instructions. Print the homework and mark the reply on the homework.
Return the hard copy of the homework marked with the replies to the instructor at the end of the class.

1) (10 points) Consider a disk with block size B = 1024 bytes, inter-block gap size G= 128 bytes, number of blocks per track = 20, number of tracks per surface = 400. The disk pack consists of 20 double sided disks. The disk drive rotates the disk pack at the speed of 2400 rpm and one track of data can be transferred per revolution.
   a) (2 points) What is the total capacity of a track, and what is its useful capacity (excluding inter-block gaps)?
   b) (4 points) Assuming that the average seek time is 30 msec, calculate the average time in msec to locate and transfer a single block, given its block address.
   c) (4 points) Calculate the average time it would take to transfer 20 random blocks and the time it would take to transfer 20 consecutive blocks using double buffering.

2) (15 points) A file has 40000 STUDENT records of fixed length. Each record has the following fields: Name (30 bytes), Ssn (9 bytes), Address (40 bytes), Phone (8 bytes), Sex (1 byte). An additional byte is used as a deletion marker. Assume an unspanned organization. The file is stored on a disk with parameters given in question 1.
   a) (2 points) Calculate the record size in bytes.
   b) (3 points) Calculate the blocking factor bfr and the number of file blocks b.
   c) (4 points) Assuming that the file is ordered by Ssn, calculate the time taken to search for a record given its Ssn value.
   d) (6 points) Calculate the average time needed to search for an arbitrary record in the file using linear search if
      i) the file blocks are stored on consecutive disk blocks and double buffering is used.
      ii) the file blocks are not stored on consecutive disk blocks.

3) (6 points) Considering the same student record format in the question 2, assume that we have a variable length record file. Also each record has a 1 byte field type, 1 byte deletion marker and 1 byte end of record marker. The file is stored on a disk with parameters given in question 1. Assuming a spanned record organization where each block has a 2 byte pointer to the next block, calculate the number of blocks needed for the file.

4) (4 points) Indicate whether the following statements are true or false.
   a) (2 points) For an XML document to be valid, the element names used in the start and end tag pairs must follow the structure specified in a separate XML Document Type Definition (DTD).
   b) (2 points) <?xml version=”1.0” and standalone=”yes”>

```
<Project>
    <Name>X
</Project>
```

   The above XML document is well-formed.

5) (15 points) Consider an internal hash table using an array of size 10. Using the hash function h(k) = k mod 10, show the result of inserting the elements 1,4,11,14,3,6,16,22,40,43 when open addressing is used for collision resolution.

6) (8 points) Consider a B-Tree of order *m*=10.
   a) (2 points) Which is the minimum number of nodes in such tree?
   b) (2 points) Which is the maximum number of nodes in such tree?
   c) (2 points) Which is the minimum number of keys that must be stored in such tree?
   d) (2 points) Which is the maximum number of keys that can be stored in such tree?

7) (14 points) Indicate for each of the following statements whether the statement is true or false.
   a) (2 points) A file can have multiple primary indexes allocated on it.
   b) (2 points) A file can have multiple clustering indexes allocated on it.
   c) (2 points) A clustering index must be allocated on a field which is a primary key for the file.
   d) (2 points) A file can have multiple secondary indexes allocated on it.
   e) (2 points) A secondary index can be allocated only on a field which has the uniqueness property for the file.
   f) (2 points) A file must always have a primary index allocated on it.
   g) (2 points) The number of entries in a clustering index is equal to the number of blocks in the data file.

8) (8 points) Consider the relation Emp of schema (Emp#, Name, Job, Salary), where Emp# is the primary key.
   a) (3 points) Specify the SQL command for creating an index on column Job of relation Emp.
   b) (2 points) Is the index created with the statement at question (8.a) a primary index or secondary index?
   c) (3 points) Specify the SQL command for dropping the index created with the previous command.

9) (20 points) Consider a disk with block size B= 512 bytes. Assume that a block pointer is P = 6 bytes, and record pointer is P_R = 7 bytes. A file has 20000 STUDENT records of fixed length. Each record has the following fields: Name (30 bytes), Ssn (9 bytes), Address (40 bytes), Phone (8 bytes), Sex (1 byte). An additional byte is used as a deletion marker.

   a) (5 points) Suppose that the file is ordered by the key field Ssn and there is a primary index on Ssn. Calculate the number of block accesses to search for and retrieve a record from the file given its Ssn value using the primary index.

   b) (5 points) Suppose that the file is not ordered by the key field Ssn and there is a secondary index on Ssn. Calculate the number of block accesses to search for and retrieve a record from the file given its Ssn value using the secondary index.

   c) (5 points) Assuming that the file is not ordered by the non-key field Name and there is a secondary index on the field Name with an extra level of indirection that stores record pointers (option 3 of section 18.1.3 in book). Also assume that there are 2000 distinct values for the field Name. Calculate the number of block accesses to search for and retrieve all records from the file that have a specific name value using the secondary index.

   d) d) (5 points) Assuming that the file is ordered by the non-key field Name and there is a clustering index on the field Name that uses block anchors. Also assume that there are 2000 distinct values for the field Name. Calculate the number of block accesses to search for and retrieve all records from the file that have a specific name value using the secondary index (assume that multiple blocks in a cluster are contiguous).