Assignment 1 (10%), CMPT-454, Fall 2025

Total mark: 85

Due date: Oct 4, 2024, 11:59pm. Submit a single pdf file to sfu.coursys.ca. Only submissions received in coursys before the deadline will be considered. Students are responsible for getting their submission into the system by this deadline. No late submission is accepted.

Estimated marking completion: Oct 14, 2024.

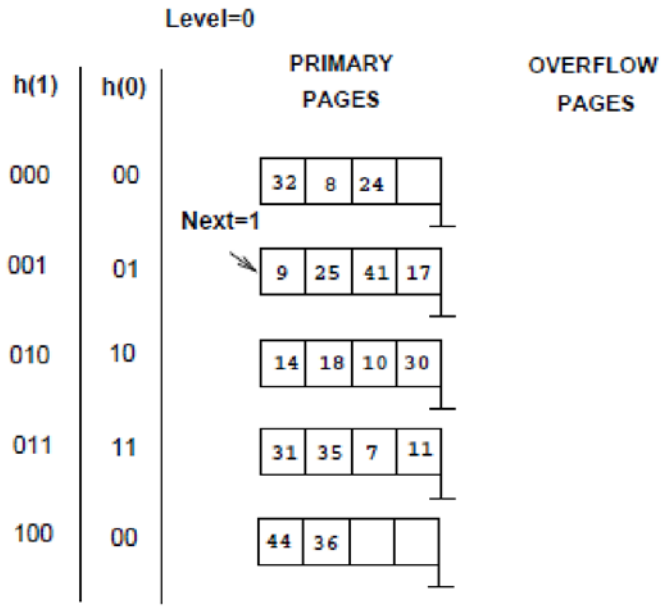
Notes: This assignment must be completed by each student independently. Violation will result in the zero mark for all parties involved. Typing of answers is required. The reasoning clarity of answers is heavily counted. A good answer should address the points in the question briefly. A longer answer does not necessarily get a higher mark. Partial mark is only for minor mistakes; major and conceptual mistakes will receive no partial mark.

Q1 (24 marks, 3 marks each). A disk page is 512 bytes. The sizes of data record, search key, rid, and page id are 40 bytes, 8 bytes, 6 bytes and 4 bytes, respectively. In the B+tree index with Alternative 2 for data entries,

1. What is the maximum number of branches for an index page
2. What is the minimum number of branches for a non-root index page
3. What is the maximum number of data entries in a leaf page
4. What is the minimum number of data entries in a leaf page
5. If the B=tree is 67% full on each node, what is the height of the B+tree to index a file of 1000,000 data records (the root has height 1).
6. What is the minimum height of the B+tree to index a file of 1000,000 data records
7. What is the minimum number of data records indexed by the B+tree with height 3
8. What is the maximum number of data records indexed by the B+tree with height 3

Q2 (10 marks). Consider a relation R(A,B,C,D). The following queries will be requested at equal frequency over time: 1) A>= a, 2) B=b, 3) C=>c, where a,b,c are constants. Assume that A>=a, B=b, C=> c have the following selectivity (i.e., the percentage of satisfying records): 20%, 10%, 5%, respectively. What indexes should be built on R to reduce the overall I/O cost of answering these queries. Justify your answers. For each index, include the information on index type (B+tree index or hash index), search key, clustered or unclustered, the alternative format for data entries. **(5 marks for correct indexes and 5 marks for information on index).**

Question 3 (12 marks, 4 marks each). Consider the following linear hash index with Alternative 2 for data entries. Assume that each page contains at most 4 data entries. Answer the following questions based on the original index.



1. What is the I/O cost of searching for records with k=44. Show the procedure of the search. Which pages are read.
2. What is the I/O cost of inserting an entry with k=20. Show the procedure of the insertion and the index after inserting the entry.
3. What is the I/O cost of inserting an entry with k=34. Show the procedure of the insertion and the index after inserting the entry?

Q4 (27 marks, 3 marks each) Answer the following questions

1. The linear hash index does not need a directory for buckets. How are buckets found without a directory.
2. In case of bucket overflow, linear hash index splits the bucket Next. How does the index reduce overflow pages.
3. Explain why extendible hash index needs a directory to buckets.
4. Describe the requirements for a clustered hash index.
5. Describe a scenario where clustered and unclustered indexes have the same I/O cost.
6. Describe when the rid of data records may change when inserting a data entry into a B+tree.

7. Explain why the “next-leaf-page” pointer in a leaf page is necessary for a B+tree but is not necessary for an ISAM index.
8. Explain why, after splitting a page A during insertion into the B+tree index, both page A and its split image page A2 still satisfy the minimum 50% occupation.
9. Explain why, after merging two pages A and B during deletion from the B+tree index, the merged page can still hold all the entries in A and B.

Q5 (12 marks, 4 marks each). Consider the following B+tree index. Assume each page contains at most 4 data records.

1. Assume that the index is clustered and uses Alternative 2 for data entries. What is the worst I/O of retrieving all data records >= 24 using this index. What is the best I/O cost of retrieving all data records >= 24.
2. Assume the index is unclustered and uses Alternative 2. What is the worst I/O of retrieving all data records >= 24 using this index. What is the best I/O cost of retrieving all data records >= 24.
3. Assume the index has Alternative 1 for data entries. What is the worst I/O of retrieving all data records >= 24 using this index. What is the best I/O cost of retrieving all data records >= 24.