COSC 404 Midterm

Key Concepts:

* Insert and Delete from B+-tree (4 marks)
* Insert with linear hashing (2 marks)
* Perform RAID and index calculations (6 marks)
* Code an iterator in Java (lab 4) (5 marks)
* Create relational query plans in Java (5 marks)

Topics:

1. Storage
   1. Storing records in memory
      1. A record consists of onme or more fields grouped together
         1. Each Tuple of a relation is a record
      2. Two main types of records:
         1. Variable length: size of the record varies
         2. Fixed length: all records are the same size
   2. Variable formats
      1. Useful cases:
         1. the data does not have a regular structure in most cases
         2. the data values are sparse in the records
         3. there are repeating fields in the records
         4. the data evolves quickly so schema evolution is challenging
      2. Disadvantages:
         1. Space is wasted by repeating schema information for every record
         2. allocating variable-sized records efficiently is challenging
         3. query processing is more difficult and less efficient when the structure of the data varies
      3. JSON & XML are best described as variable format, variable size
      4. A VARCHAR field is best described as fixed format, variable size
   3. Storing records in blocks
      1. Issues related to storing records in blocks
         1. Separation: how are adjacent records separated?
            1. variable length records can be separated by:

a special separator marker in the block

storing the size of the record at the start of each record

store the length or offset of each record in the block header

* + - * 1. Spanning

unspanned: do not allocate records across blocks and waste space

Blocks needed for unspanned records = 1record/block

utilization = recordSize/blockSize\*100%

spanned: start a record at the end of a block and continue on to the next (round up)

each piece must have a pointer to its other part

blocks needed = numRecords\*recordSize/blockSize

* + - 1. Spanning: can records cross block boundaries?
      2. Clustering: how many records can a block store?
      3. Splitting: how many blocks are records allocated in?
      4. Ordering: are the records sorted in any way?
      5. Addressing: how each record is referenced

1. Indexing
2. B-Trees
3. R-Trees
4. Hashing
5. SQL Indexing
6. Query Processing
7. Query Optimization