1: B+ Tree Indexing

Consider the B+ tree index of degree \( d = 2 \) shown in Figure 1. Instead of pointers to the data records in data files, the leaf nodes in this B+ tree contain data records. The insertion, deletion, and update algorithms for this B+ tree is the same as the ones discussed in the lecture.

1. Show the tree that would result from inserting a data entry with key 9 into this tree.
2. Show the B+ tree that would result from inserting a data entry with key 3 into the original tree.
3. Show the B+ tree that would result from deleting the data entry with key 8 from the original tree, assuming that the left sibling is checked for possible redistribution.
4. Show the B+ tree that would result from deleting the data entry with key 8 from the original tree, assuming that the right sibling is checked for possible redistribution.
5. Show the B+ tree that would result from starting with the original tree, inserting a data entry with key 46 and then deleting the data entry with key 52.
6. Show the B+ tree that would result from deleting the data entry with key 91 from the original tree.
7. Show the B+ tree that would result from starting with the original tree, inserting a data entry with key 59, and then deleting the data entry with key 91.
8. Show the B+ tree that would result from successively deleting the data entries with keys 32, 39, 41, 45, and 73 from the original tree.

2: B+ Tree Indexing

The degree of a B+ tree (d) can be a non-integer value. If the value of d is not integer, the minimum and maximum number of keys in the internal nodes will be \( \text{floor}(d) \) and \( \text{floor}(2d) \), respectively. The root is the exception and can have between 1 and \( \text{floor}(2d) \) keys. Consider the B+ tree shown in Figure 2, where the value of d is 1.5, so each internal node has between 1 and 3 keys.
Oregon State University Practice problems 2

CS 440, Winter 2017

Figure 2: The original B+ tree for question 2.

1. Delete keys 13 and 17 from this B+ tree.

2. Insert keys 49, 51, 55 and 60 into this B+ tree.

You only need to show the final picture of the B+ tree.

3: B+ tree Indexing

Using B+ trees with degrees of two (2) whose keys are integer values, answer the following questions. A B+ tree with a single node has height of 1.

1. Give an example of a B+ tree whose height changes from 2 to 3 when the value 25 is inserted. Show your structure before and after the insertion.

2. Give an example of a B+ tree in which the deletion of the value 25 causes a merge of two nodes but without altering the height of the tree.

4: B+ tree Indexing

To reduce the number of I/O access in index search, each B+ tree node should fit in a block. Let the key value and record pointer for a B+ tree be 32 and 64 bytes, respectively. If the block size is 16384 bytes, what should be the minimum degree of the B+ tree?