CMSC 420: Homework 3: B+-Trees

- Handed out: 4/23 Due: 4/30 before class starts. Late homeworks will not be accepted.
- Optional questions do not carry points but may be important for “grade-fence sitters.” Answer optional questions on a separate piece of paper to be retained by the instructor. Otherwise, please answer in the space provided.

1. (5 points) The definition on page 9 has a strict inequality. Show (by means of an example using $t = 4$) that our pseudocode does not enforcing this inequality. Specifically, give a sequence of b-tree-insert(), or search() or remove() calls.

2. (25 points) The sequence 90,80,10,30,70,20,50, 92,82,12,34,73,24,55, 91,81,11,31,71,21,57 is inserted in order in a B+-tree with $t = 3$. We then issue FIND(70).

   (a) Show two trees: the one after INSERT(55) and the tree after all insertions.

   (b) How many DiskRead() calls will be made for FIND(70)? Assume root disk page cached in memory.

   (c) Show the tree after the 10 largest items are removed.
3. (20 points) What is the maximum number of keys a B+-tree can have as a function of the height $h$ and branching factor $t$? (Each node can have at most $2t - 1$ keys as usual.) Explain.

4. (10 points) Give pseudocode (in the manner of the notes) for $\text{rotateL}$ and $\text{rotateR}$. 