B+ tree

Read B+ tree indexing from the Raghu Ramakrishnan's book.

Our B+ tree will have two parameters: d and t

- A leaf or data node has capacity to store 2d records. Each leaf node should contain at least d records, except when there is only one leaf node in the B+ tree.
- A non-leaf or index node has capacity to store 2t+1 keys. Each non-leaf node should contain at least t keys, except when there is only one non-leaf node in the B+ tree.

In this assignment, you are supposed to handle only insertion of records.

For handling insertion of records, you have to only implement the node splitting strategy discussed in the class.

Example input 1:

First two lines describe the initialization parameters for your B+ tree: d and t.

Line 1: d=2. Each data node will have capacity to store 4 records.

Line 2: t=1. Each index node will have capacity to store 3 keys.

Line 3: Operation code 1 indicates insertion of value into the B+ tree. The value to inserted is given after a blank space. Assume that the value to be inserted is integer.

Line 7: Operation code 2 indicates displaying of status of your B+ tree. You are expected to mention only the number of index and data nodes, followed by the contents of the root node.

Line 23: Operation code 3 indicates end of input. Your program should quit.

We will be following conventions:

- Values in the nodes are arranged in the ascending order from left to right.
- There are no duplicates to handle.
- When a data node splits, it creates two data nodes: left node and right node. The left node will have d records. The right node will have d+1 records. Separator sent to the parent node will be the smallest record stored in the right node.
- When an index node splits, it creates two index nodes: left node and right node. The left node will have t keys. The right node will have t+1 keys and one key is sent to the parent node.

Example output 1:

Line 1: After inserting first four values (100, 200, 300, 400), the B+ tree has zero index nodes and one data node. Current data node is the root node for the B+ tree.

Line 2: After inserting 500, the B+ tree has one index node and two data nodes. The index node is the root node and it contains a single key: 300.

Line 3: After further inserting 600 and 700, the b+ tree has 3 data nodes and 1 index node. The index node contains keys 300 and 500.

Similarly, you can interpret the rest of output file.

Note on code submission:

Make sure that all your code fits into a single file <roll number>Bplus.cpp

How your code should compile?

g++ <roll number>Bplus.cpp

How your code should run?

a.out < input_file_name

Your code should write the output on the stdout