18557 - Akhilesh

/* Evaluation:

1. Code compilation:

1. Does code compile without errors? - Yes.
2. Was a readme.txt file included with instructions on how to compile and run?

2. Correctness (positive test cases):

1. Can I insert a key?
2. Can I delete a key? - No.
3. Can I search for a key? - No
4. Can I view display of tree? - Not in the manner requested (graph. The textual display is confusing).
5. Can I specify size of B+ tree node (# of keys in a node)? - Not as a user.
6. Do the nodes satisfy the B+ tree property? - Unable to verify since display as graph is not supported.
7. Can I create a B+ tree from a file of keys? - yes.
8. Can I save my B+ tree to a file? - No
9. Can I load back the file saved in step 7? - No.
10. Can I insert and delete keys from the command line even after loading keys from file? - No.
11. Is Output for keys1.txt correct? - Unable to verify
12. Is output for keys2.txt correct? - Unable to verify.

3. Programming Style & General Comments:

1. Are there useful comments that complement the code? - Partially.
2. Is the indentation style neat and consistent? - yes
3. Are there had coded limits or magic numbers used in the code? - Yes. Only the first 25 characters are accepted as key:
   BPT.java line 62: `if(s.length()<25)`
   ```
   Also, number of pointers in a node is hard coded:
     Page.java line 9 - public int order= 6; //no of pointers i.e., order-1 key values should be there
   ```
4. Are there hard coded file paths used in the code? - No.

4. Exception Handling:

1. Delete on empty tree. - Delete not implemented.
2. Delete a non-existent key - Delete not implemented
3. Insert a key that exists already (keys3.txt) - Works okay. Duplicates are allowed.
4. Call display on empty tree(keys5.txt) - handled.
5. Print an empty tree(keys5.txt) - handled.
6. Empty lines in input file (keys4.txt) - not handled. Empty lines are treated as keys.

Score - 10/20

*/