18559

/* 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? - The readme document is well written. Thank you!

2. Correctness (positive test cases):
  1. Can I insert a key? - only from input file.
  2. Can I delete a key? - not implemented
  3. Can I search for a key? - yes.
  4. Can I view display of tree? - Yes. Automatically
  5. Can I specify size of B+ tree node (# of keys in a node)? - Yes. Not on the command line though.
  6. Do the nodes satisfy the B+ tree property? - Yes
  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? - Yes
  12. Is output for keys2.txt correct? - Yes

3. Programming Style & General Comments:
  1. Are there useful comments that complement the code? - Minimal comments provided.
  2. Is the indentation style neat and consistent? - yes.
  3. Are there had coded limits or magic numbers used in the code? - Yes. Key size is limited to 15 characters in Btree.java line 112:
```
   b.readFile(15);
```

  4. Are there hard coded file paths used in the code? - yes.
    The file name for file containing keys is hard coded. This could have been avoided easily!
  5. General Comments:
    a. I liked the addition of a file for global constants. Next time, try to have such values accepted from a configuration file user can provide.

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) - Does not allow duplicate keys.
  4. Call display on empty tree - workd
  5. Print an empty tree - Works.
  6. Empty lines in input file (keys4.txt) - Empty lines seem to be stripped out. Empty spaces before keys on a line are not getting stripped properly.

Score - 13/20.

*/