# 18558 - Saiteja

/* 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? - yes. The readme.txt file was well written!

**2. Correctness (positive test cases):**

1. Can I insert a key? - yes.
2. Can I delete a key? - yes. It does not work always though. It seems that only keys added after loading a file are getting deleted (at least as per the graph displayed).
3. Can I search for a key? - No. There is a search option. It does not work correctly. For example, I get a key not found message when searching for key A after loading keys2.txt. The key is present in the file and is also shown in the graph.

The search option does not provide a way to return to the main menu.

```
Choose your option:
's': to search a key
'p': to print the tree graph
'd': to delete a key
'i': to insert a key
'w': to write to a file
'r': to read from a file
'e': to exit
Enter:  s

Enter a Key: A
Key: "A" does not exist

Enter a Key: C
Key: "C" does not exist

Enter a Key:
Key: "" does not exist
```

4. Can I view display of tree? - Yes. The program expects the presence of file named graph.png always. It throws an exception if file is not present.

```
Exception in thread "main" java.lang.IllegalArgumentException: The file: graph.png doesn't exist.
at java.awt.Desktop.checkFileValidation(Desktop.java:208)
at java.awt.Desktop.open(Desktop.java:267)
at B_plusTree.print_graph(B_plusTree.java:665)
at B_plusTree.main(B_plusTree.java:705)
```

5. Can I specify size of B+ tree node (# of keys in a node)? - Yes
6. Do the nodes satisfy the B+ tree property? - Yes for the values I tested.
7. Can I create a B+ tree from a file of keys? - yes.
8. Can I save my B+ tree to a file? - yes.
9. Can I load back the file saved in step 7? - yes.
10. Can I insert and delete keys from the command line even after loading keys from file? - only delete.
11. Is Output for keys1.txt correct? - nearly correct.
12. Is output for keys2.txt correct? - nearly correct.

**3. Programming Style & General Comments:**

1. Are there useful comments that complement the code? - No
2. Is the indentation style neat and consistent? - yes
3. Are there had coded limits or magic numbers used in the code? - Yes.
   length of key seems to be hard coded in B_plusTree.java
   ```
   if((length=st.length()) > 4)
      key = st.substring(0,4);
   ```
4. Are there hard coded file paths used in the code? - yes. The output graph file name is hard coded:
   ```
   String dot_file="graph.dot";
   ```

**4. Exception Handling:**

1. Delete on empty tree - unable to verify since delete is not deleting from tree!.
2. Delete a non-existent key - handled.
3. Insert a key that exists already (keys3.txt) - works. Duplicate keys allowed. .
4. Call display on empty tree(keys5.txt) - handled.
5. Print an empty tree (keys5.txt).
6. Empty lines in input file (keys4.txt) - not handled. Empty lines are being treated as keys.

Score - 13/20.

*/