Semester Project report

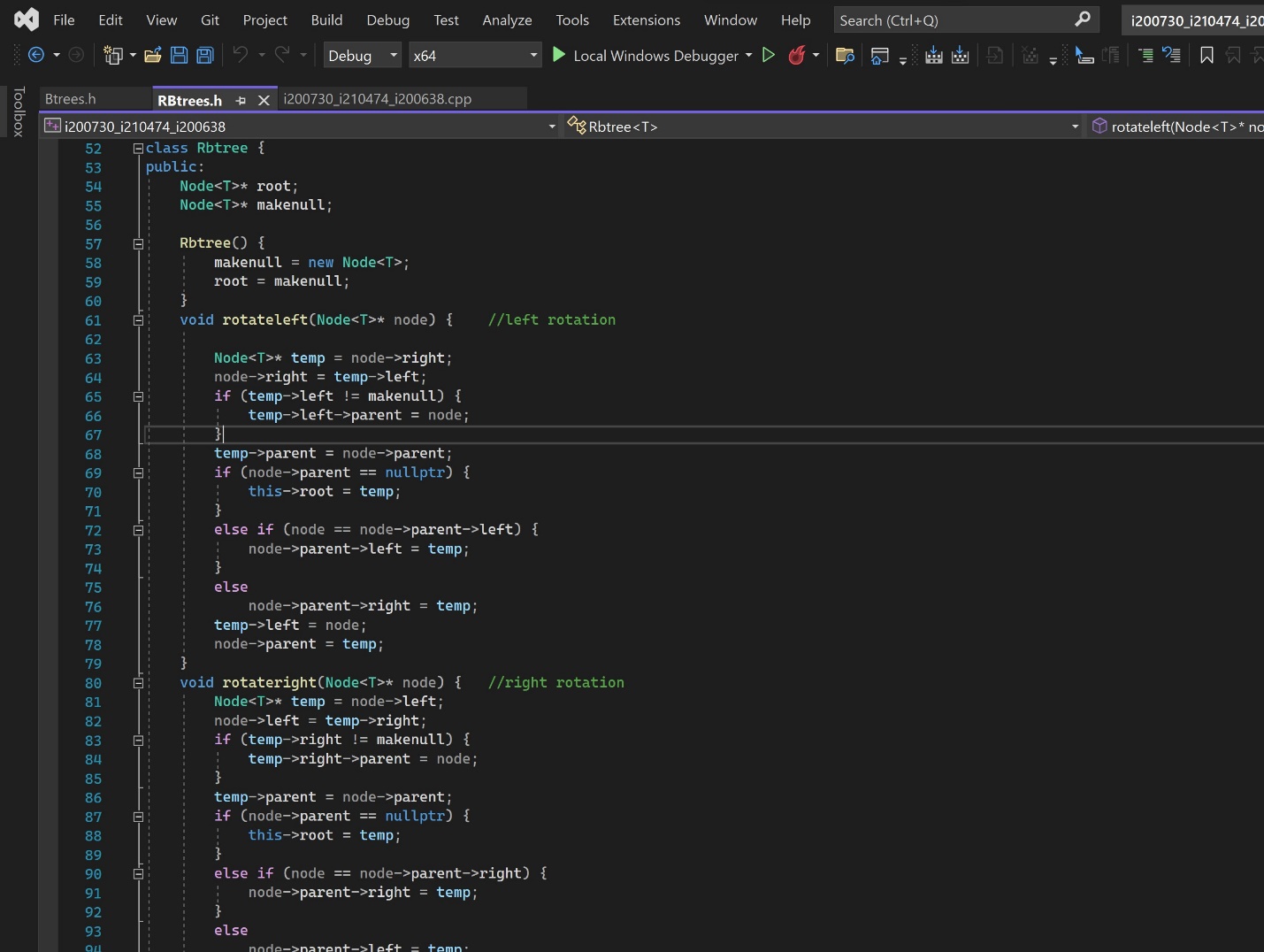
20i-0730 Bilal Ahmad

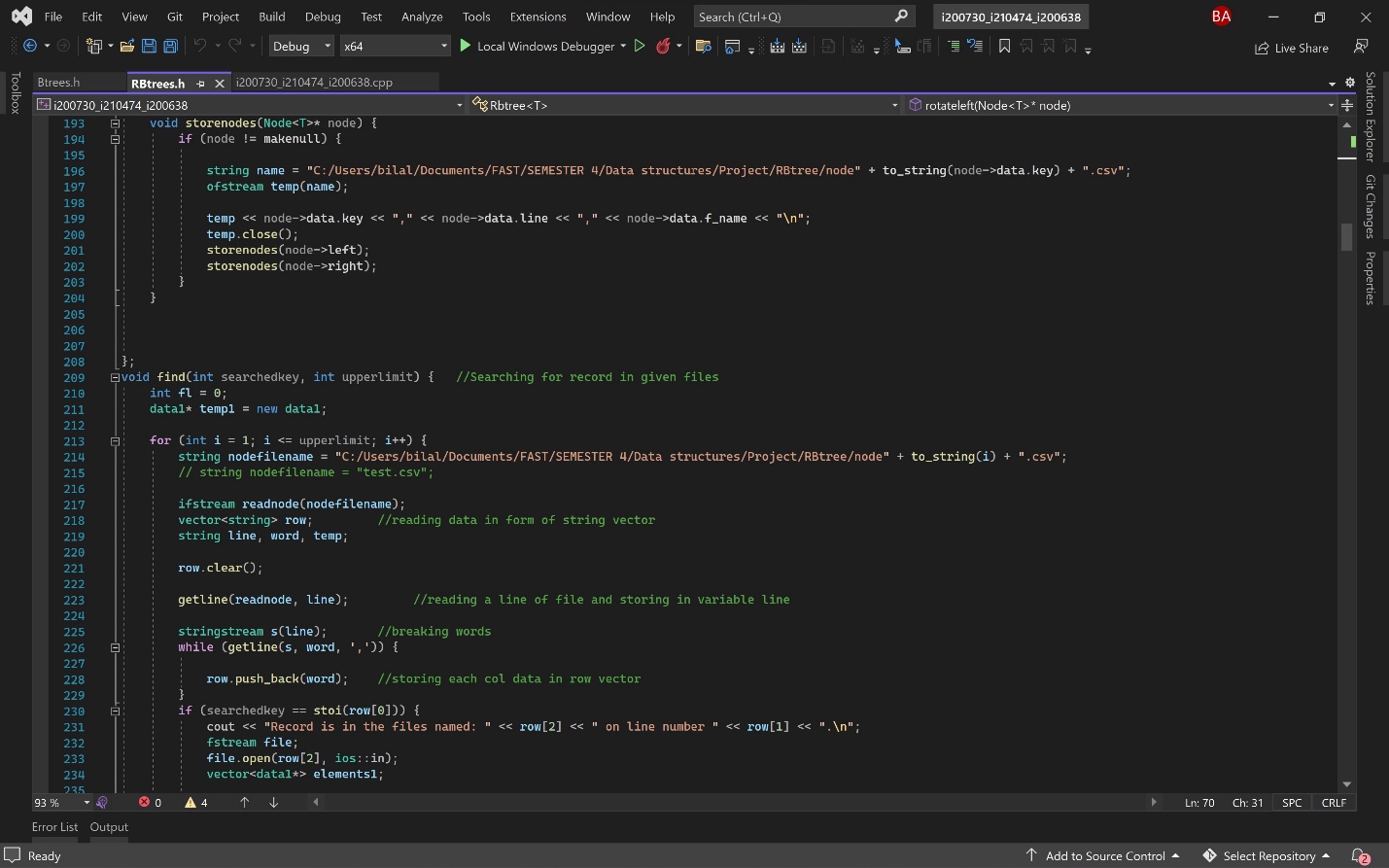
21i-0474 Muhammad Affan

20i-0638 Muhammad Maaz

# RB trees

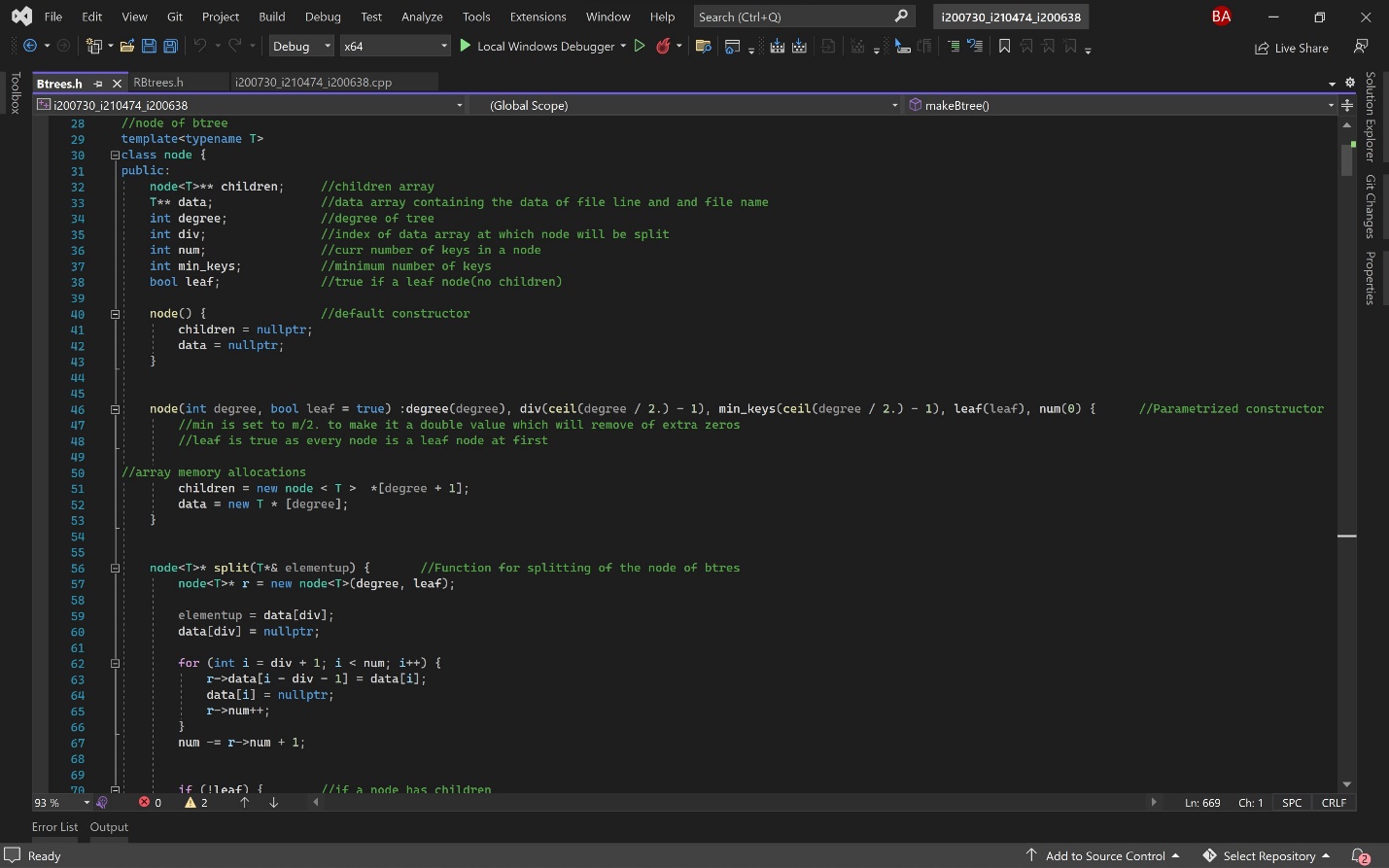
We made a separate class for data for RBtrees to use as a template argument, for the data to be stored in thenodes of tree. Here in the RB tree a node is made containing the data variable of type ‘Data1’ which is declared above.

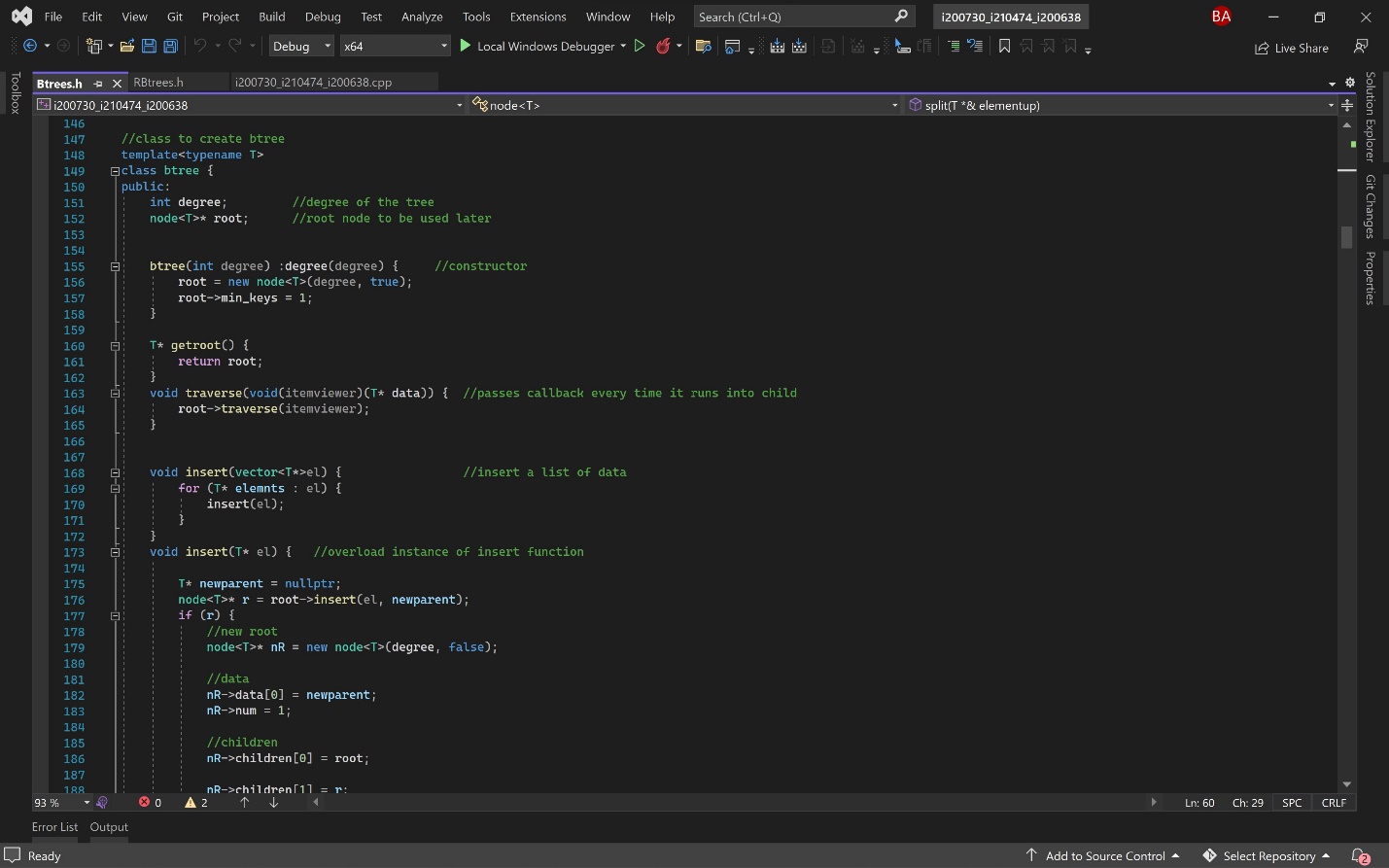


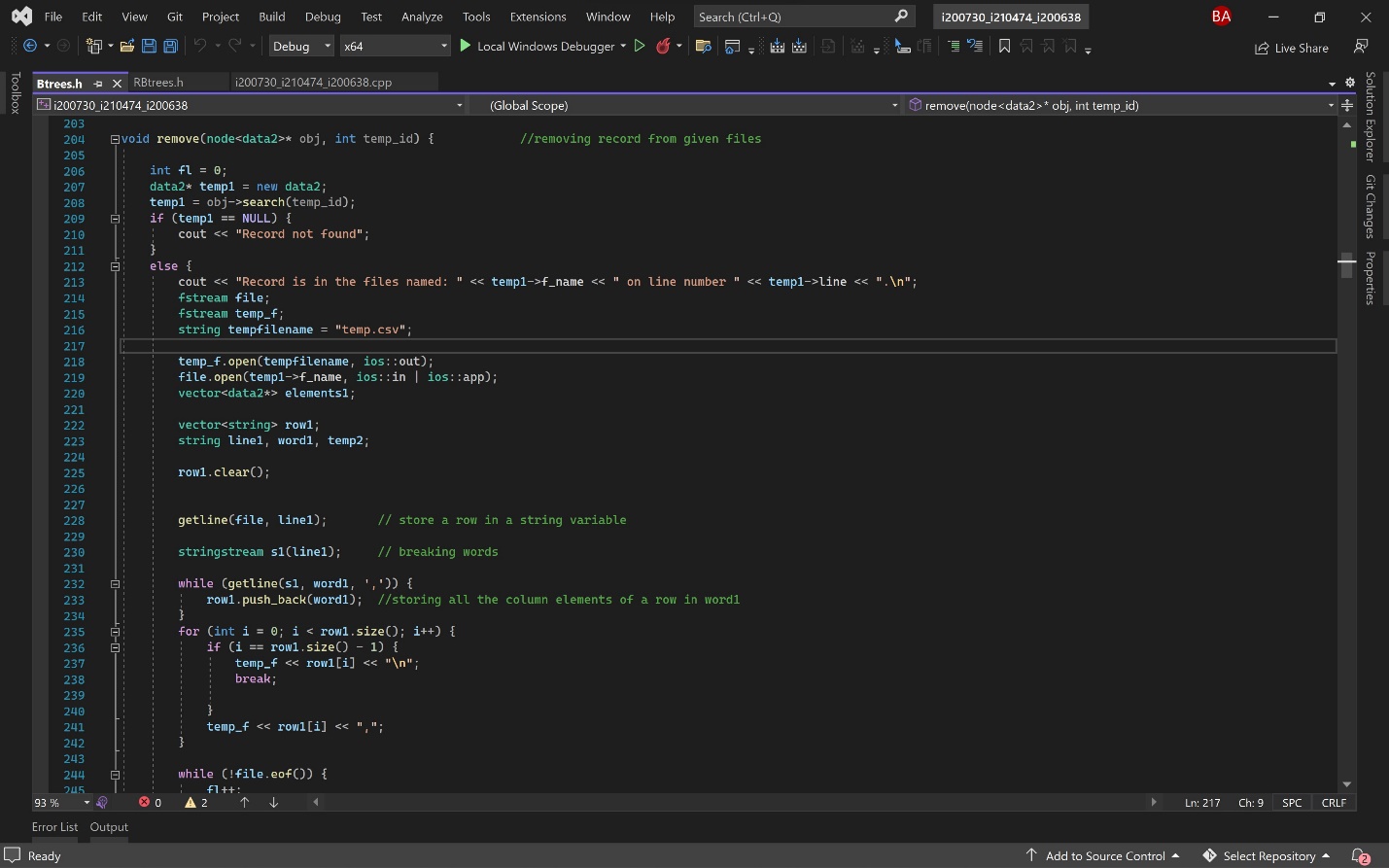
Then a class for tree is made containing the root pointer of Node type and a nullnode pointer of Node type as well. The keys are read from files and stored he re in the nodes for the making of tree. 

Then the nodes created are stored in separate files so if the program is restarted the tree does not to be made again. There are three separate functions of find, delete and update a specific tuple in the data files give. The files are first red to make the nodes. The odes contain the key, line number and filenames of the tuples. A sperate file is made which contains the roots and the ranges of each tree. Then when the queries are run the file with the data for root is opened to match the key and for the actions to be performed. The RB trees support runtime updating as well.

# B trees



Similarly, for Btrees the data2 structure is made to store the data and be used as a template arguments later. The node class here contains 2d pointers as the nodes of Btrees are them selves arrays hence there is node pointer pointing to an array which itself is a pointer.The class for the tree contains the root node and the variable to store the degree of the tree. Along with this it has usual helper functions to help with the process of insertion, searching etc.

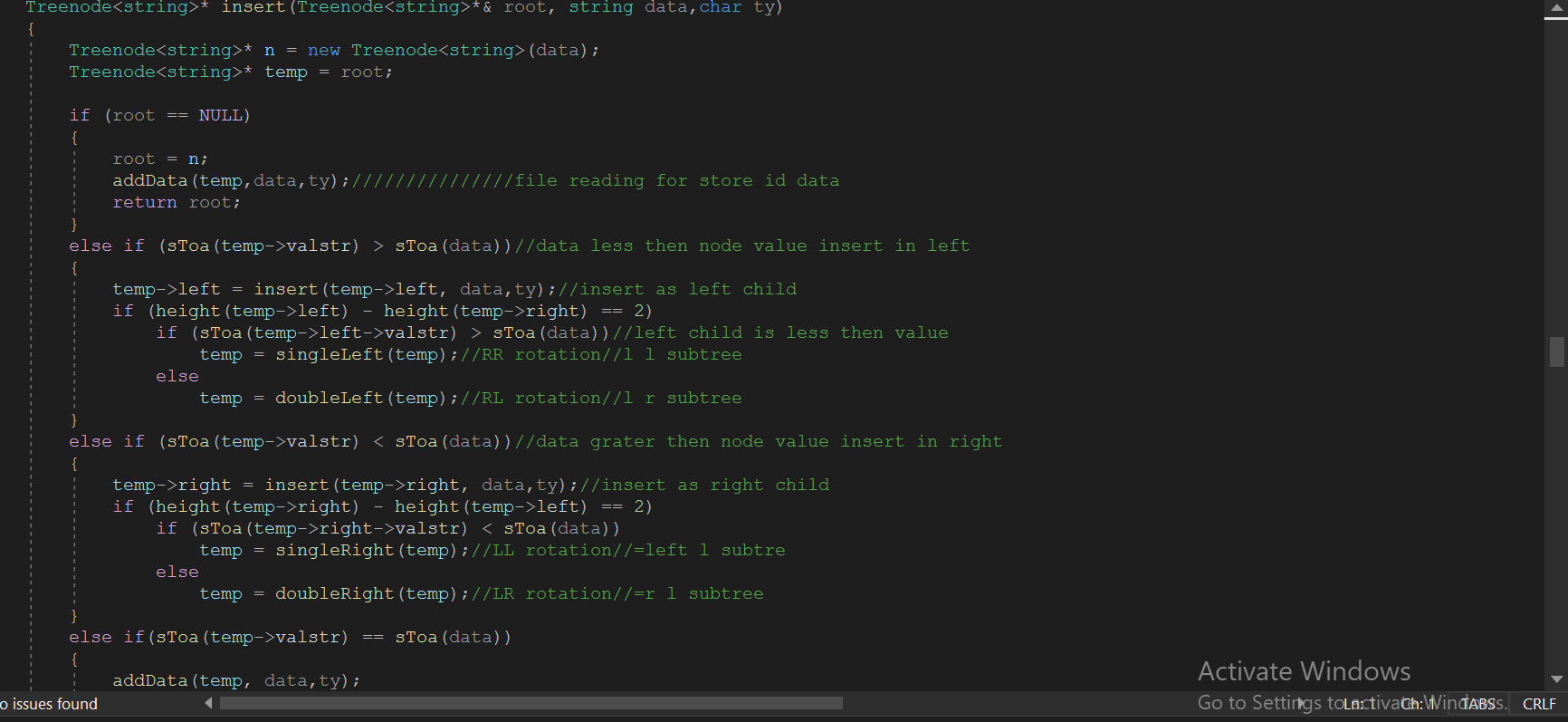


Similar to the Rb trees the B trees also have separate functions for the process of deletion, searching and updating of trees. Which searches through the tree and for deleting and updating, all the data except for the tuple to be deleted or updated is copied to a new file which is then renamed.

Text

Description automatically generated**AVL**

Similarly, in AVL tree firstly data is taken input and then balanced by using LL, LR, RL,RR rotation technique and then balance. ID is used as key in it. ID is unique so to find Data is much easier in it but data can be found by other searches too. Moreover, data sorted is also stored in that current file.



Separate Functions are made to insert, delete, update and also find data. When the search is being conducted. Firstly, sorting is being processed and the file from which data is taken the same file is used to update and delete qeuery.

Graphical user interface, application

Description automatically generated

Level Order Traversal is used as we found it much suitable to resolve problems and guess that our traversal is truning correctly.