|  |
| --- |
|  |
| **ICS202 Project** |
| SEC:52  Hassan ali alzaid  201943850 |

**In the project I use the BT data starcher and I implement** **Dictionary class witch contains Word Pare nodes I implements four classes(WordPare, Dictionary,** **BinaryTree,** **TestClass) in this report ill explain them.**

**Class** **WordPare:**

**WordPair(T word,T meaning)**

Decleration of word pare node that takes String word and meaning and store it in the node

Big-O is O(1)

**WordPair(T word,T meaning, WordPair<T> left, WordPair<T> right)**

link word pare nodes with each ether

Big-O is O(1)

**String toString()**

Big-O is O(1)

**Class** **Dictionary:**

**Dictionary(WordPair root)**

Big-O is O(1)

**Dictionary(String filePath)**

This method will read the file if its exists and put each word and their meaning in a Wordpare node and put it in the Dictionary.

If the file is exists, Big-O is O(n)

If not Big-O is O(1)

**boolean isEmpty()**

This method will find if the dictionary is empty or not

Big-O is O(1)

**int getSize()**

This method will count the number of word in the dictionary.

Big-O is O(n)

**boolean insert (WordPair wordpair)**

Big-O is O(logn)

**WordPair find(String word)**

Big-O is O(logn)

**boolean delete(String word)**

This method will delete a WordPare node form the dictionary that contains the selected word

Big-O is O(logn)

**boolean modifyWord(String word, String newMeanings)**

This method will find a word then replace its meaning with new one.

Big-O is O(logn)

**void printAll(String prefix)**

this method will print all the word that start with matching prefix and their meaning in the Dictionary.

Big-O is O(n)

**void printSorted()**

this method will print all the word and their meaning in the Dictionary.

Big-O is O(n)

**String saveString()**

this method will return string all the word and their meaning in the Dictionary.

Big-O is O(n)

**void DictionaryClose()**

This method will write the edits in the file all the words and their meaning in in the Dictionary if the file exists.

If the file is exists, Big-O is O(n)

If not Big-O is O(1)

**Class BinaryTree:**

**boolean isEmpty( )**

this method will check if the tree is empty or not by checking the root.

Big-O is O(1)

**void insert(T key, T Mean)**

insert node in the tree

Big-O is O(logn)

**void deleteByCopying(T data)**

delete selected node

Big-O is O(logn)

**void inorderTraversal()**

this method will print all the tree nodes in the tree in ordered.

Big-O is O(n)

**boolean search(T key)**

**Class TestClass:**

Contains only the main method, where we start the program.