Cairo University
Faculty of Computers and Information

Data Structure

Project Report

Name	ID
Ahmed Nasser Anwer	20110098
Diaa Mohsen	20110213
Abdul Rahman Abdul Moneim	20110220

Task 1

In Task 1, we make 4 classes Node , Linked list ,contact and contact manger to make linked list contact , contact contain linkedlist of phone number and string of name this simple description of project

we implement function this loading from File truly name and phone number that sent make new contact and send it to function add in contact manger this function put name and phone in contact and send to function addcontact that add it in linkedlist of contact manger

Also, We handled entering names sorted in a Linked List, but we faced a problem in entering multiple phones for one contact name but we finally we handled it.

For “cout” operator we substituted this operator with a function called print() in Contact Manager, but we finally handled it.

The relations and operations simply is finalized in addSorted function in Linked List, name is loaded from file, then function add in contact Manager is called, and then an object is created from contact, function addSorted is called.

Task 2

In task 2 we make Treap : It is same types of trees, including AVL trees and red-black trees , that help us to short number of search in linked list , This algorithm is very similar to quicksort. Picking the element with the maximum priority key is equivalent to picking out a random element. Then we identify the elements which are less than and greater than the selected element.

We assume max priority key takes linear work and logarithmic depth. So this algorithm, like quicksort, has O(n log n) expected work and O(log2 n) depth with high probability. Since we proved that the recursion depth of quicksort is O(log n) with high probability, this means that build Tree has O(log n) recursion depth with high probability, and hence a treap has depth O(log n) with high probability.

Our function is : insert that use rotate right and left function use in priority swap and find that search in Treap about key and find the value of it.

Finaly we sub it to Treap.h and Treap.cpp

Task3

We make Trie to search in my contact name using sub first character this application found in android that Looking up data in a trie is faster in the worst case, O(m) time (where m is the length of a search string), we convert every string of characters to string of numbers to make searching easier. Inserting every element in trie as same as inserting characters. We search it using same way, take number and search for it in the trie, we present all possible combination by using function.