Assignment 5

Submitter's Details :

Name: Karanpreet Singh

Enrollment Number : 18114036

Class : B.tech , CSE 2nd year

Email: ksingh2@cs.iitr.ac.in

Phone – 9530894998

Github repository : https://github.com/karan0299/CSN-261-ASSIGNMENT

Question 1

Write a C++ program to perform addition and multiplication of two polynomial expressions using any data structure chosen from STL. The polynomial expressions are of the form ax2 + bx + c, where a, b and c are real constants. The inputs for 2x2 + 5x + 6 and 2x3 + 5x2 + 1x + 1 are shown below (real constants followed by their power of x).

DataStructures :

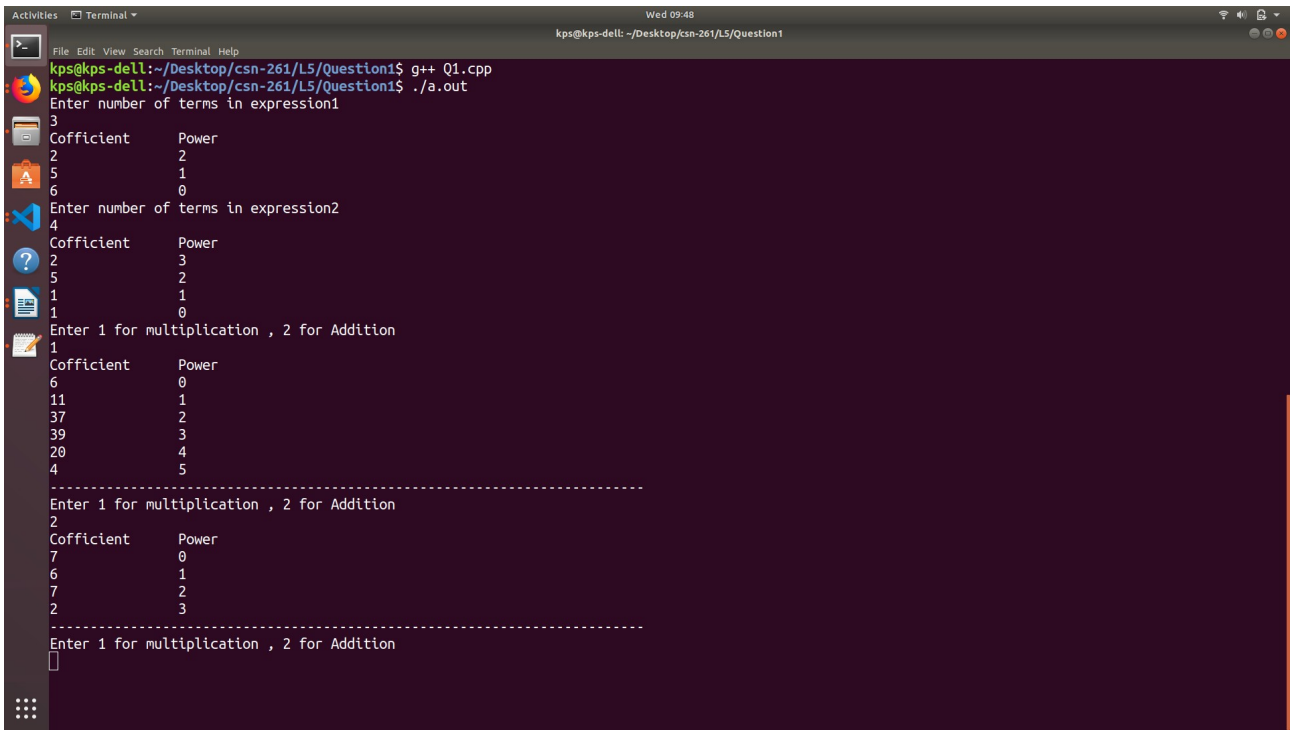
i) Ordered Map

Maps are associative containers that store elements formed by a combination of a *key value* and a *mapped value*, following a specific order.

Algorithm:

Cofficients are mapped to corresponding powers.The result of Substraction and Addition also stored in map.

ScreenShots:



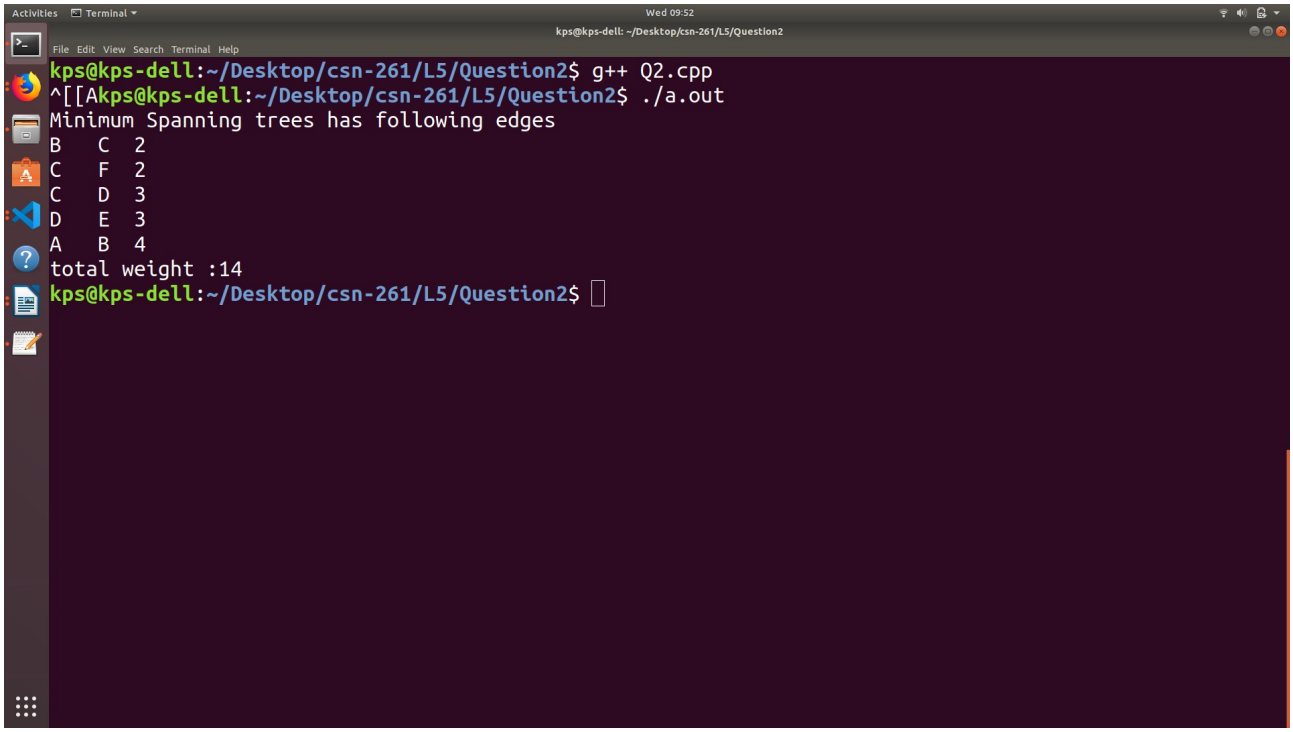
Question 2

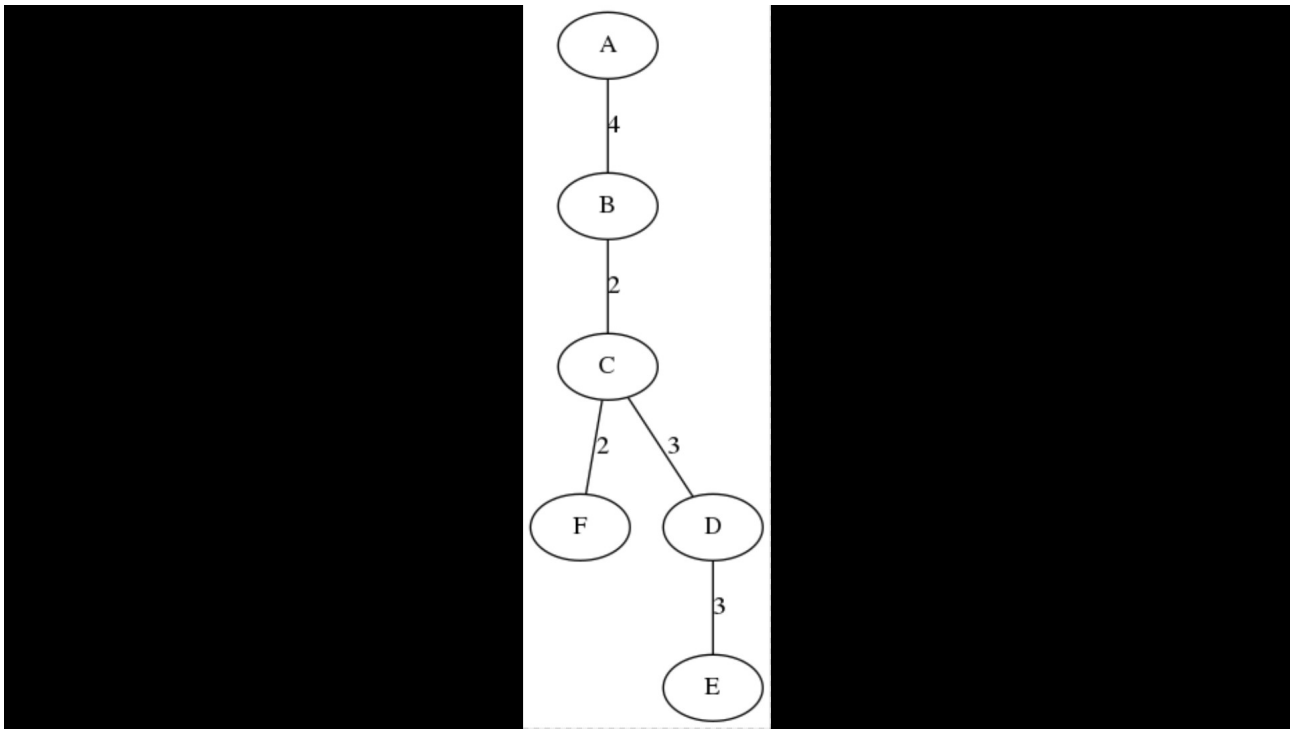
Given a set of nodes connected to each other in the form of a weighted undirected graph G, find the minimum spanning tree (MST). A spanning tree T of an undirected graphG is a subgraph that is a tree which includes all of the vertices of G, with minimum possible number of edges. G may have more than one spanning trees. The weight of a spanning tree is the sum of weights given to each edge of the spanning tree. A minimum spanning tree (MST) is a spanning tree whose weight is less than or equal to that of every other spanning tree.
For given input graph (given as a CSV file having the format as shown in the example below), implement Kruskal's algorithm in C++ program using UNION FIND data structures (without using STL) and show all the edges of the MST as output in both the command line and in the "dot file", where DOT is a graph description language. Also, print the total edge weight of the MST. Further use the "dot file" file to visualize the output graph in .pdf or .png file using Graphviz.

DataStructures :

i) *Vector*
ii) *Set*
iii) *UNION FIND data structure* : union–find data structure is a data structure that tracks a set of elements partioned into a number of disjoint (non-overlapping) subsets. It provides near-constant-time operations to add new sets, to merge existing sets, and to determine whether elements are in the same set.

ScreenShots:





Algorithm: Kruskal's Algorithm

This algorithm is a minimum-spanning-tree algorithm which finds an edge of the least possible weight that connects any two trees in the forest.It is a greedy algorithm

in graph theory as it finds a minimum spanning tree for a connected wieghted graph adding increasing cost arcs at each step.

Question 3

Write a C++ program to implement Prim's algorithm for a given input graph (given as a CSV file having the format as shown in the example below) using Fibonacci heap data structure to find the minimum spanning tree (MST). You can use STL for the data structure used in this C++ program.

DataStructures :

i) *Vector*

ii) *Set*

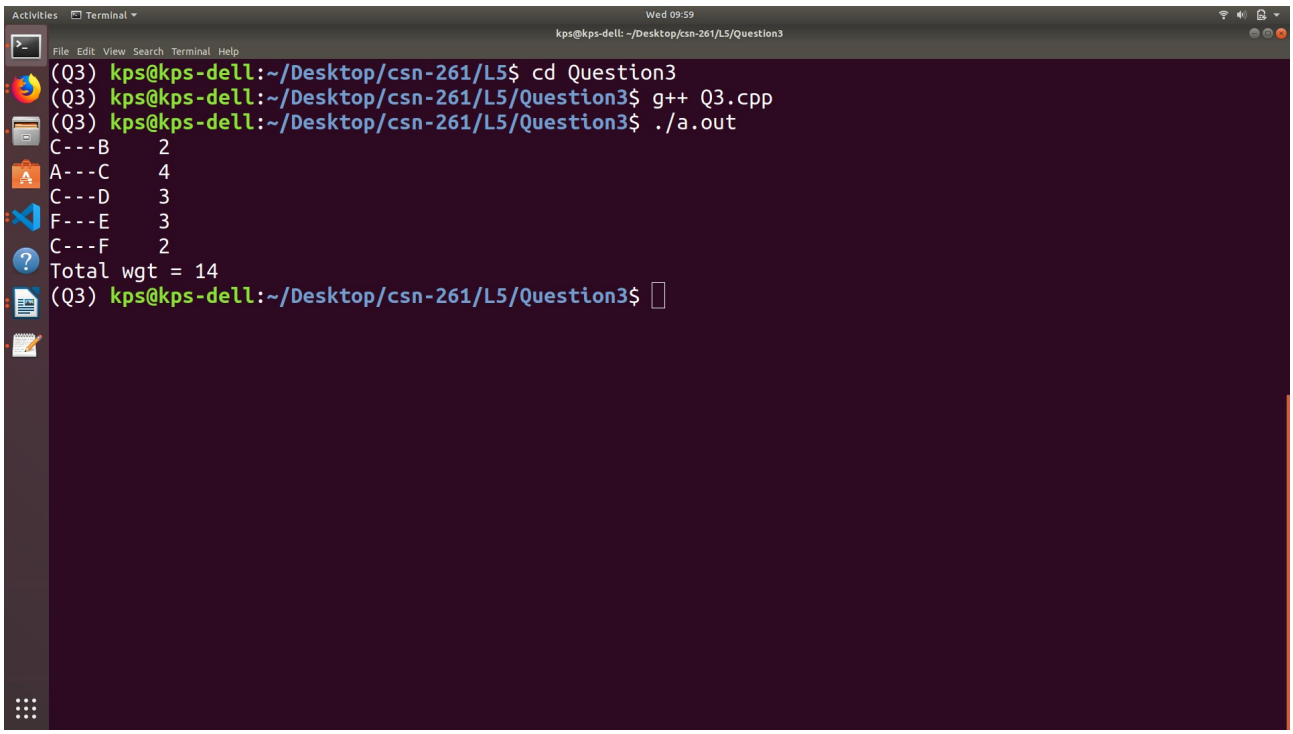
iii) Map

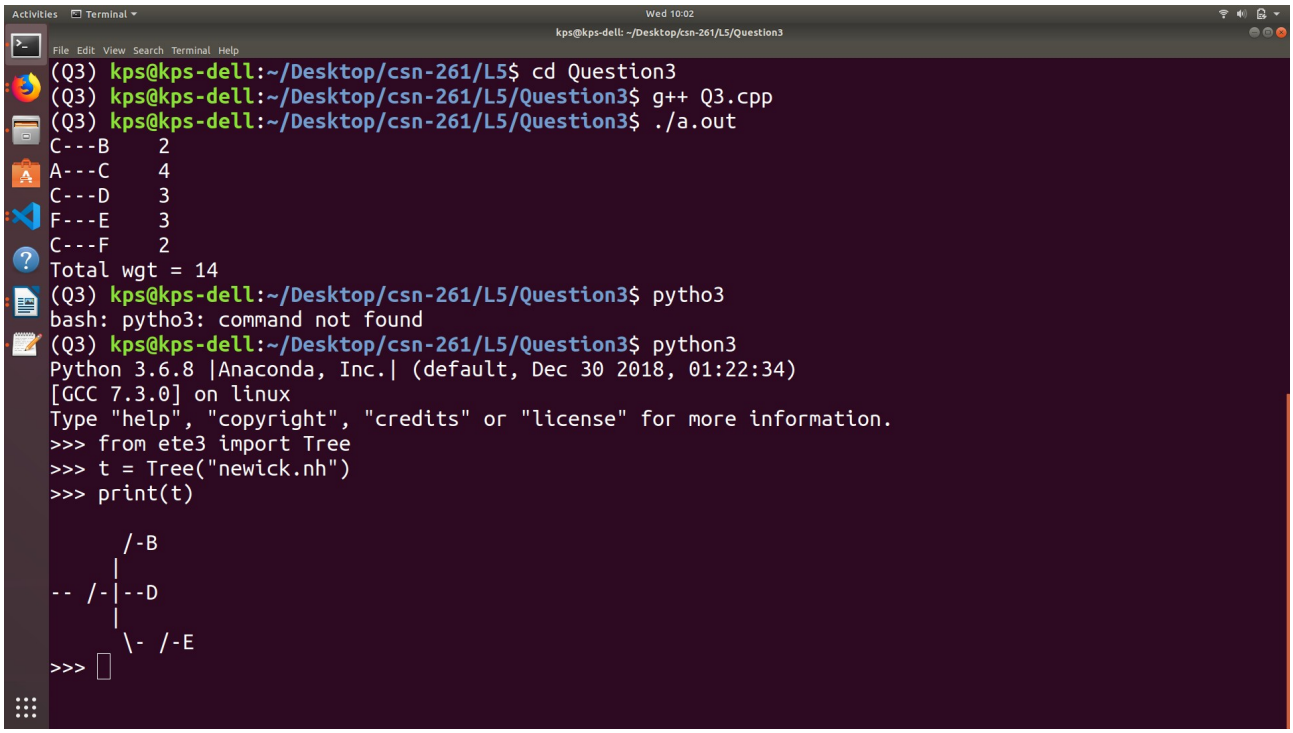
iii) *Fibonacci Min Heap* : **Fibonacci heap** is a data structure for priority queue operations, consisting of a collection of heap-ordered Trees . It has a better amortized running time than many other priority queue data structures including the binary heap and binomial heap .In Fibonacci Heap, trees can can have any shape even all trees can be single nodes.

Algorithm: Prim’s Algorithm

Prim's (also known as **Jarník's**) **algorithm** is a greedy algorithm that finds a minimum spanning tree for a weighted undirected grap h . This means it finds a subset of the edges that forms a trees that includes every vertex, where the total weight of all the edges in the tree is minimized.

ScreenShots:


Activities
Terminal
Wed 09:59
kps@kps-dell: ~/Desktop/csn-261/L5/Question3
File Edit View Search Terminal Help
(Q3) kps@kps-dell:~/Desktop/csn-261/L5$ cd Question3
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$ g++ Q3.cpp
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$ ./a.out
C---B 2
A---C 4
C---D 3
F---E 3
C---F 2
Total wgt = 14
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$


Activities
Terminal
Wed 10:02
kps@kps-dell: ~/Desktop/csn-261/L5/Question3
File Edit View Search Terminal Help
(Q3) kps@kps-dell:~/Desktop/csn-261/L5$ cd Question3
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$ g++ Q3.cpp
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$ ./a.out
C---B 2
A---C 4
C---D 3
F---E 3
C---F 2
Total wgt = 14
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$ pytho3
bash: pytho3: command not found
(Q3) kps@kps-dell:~/Desktop/csn-261/L5/Question3$ python3
Python 3.6.8 |Anaconda, Inc.| (default, Dec 30 2018, 01:22:34)
[GCC 7.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from ete3 import Tree
>>> t = Tree("newick.nh")
>>> print(t)
/-B
|
-- /-|--D
|
\- /-E
>>>