ASSIGNMENT 5

Name : Harshit Verma

Enrollment number : 18114029

Email : hverma1@cs.iitr.ac.in

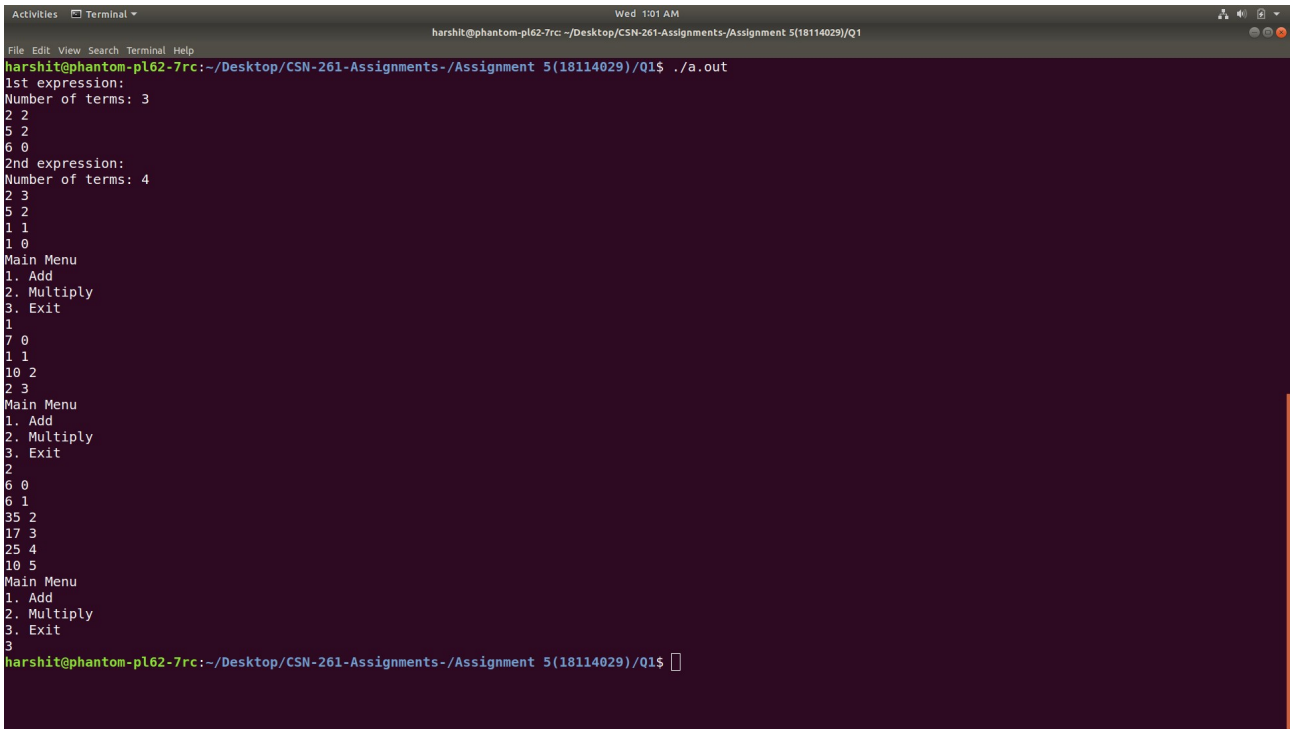
Phone-number : +91 7905362294

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).

Algorithm: nil
Data-structure: Hashmap

Screenshots:

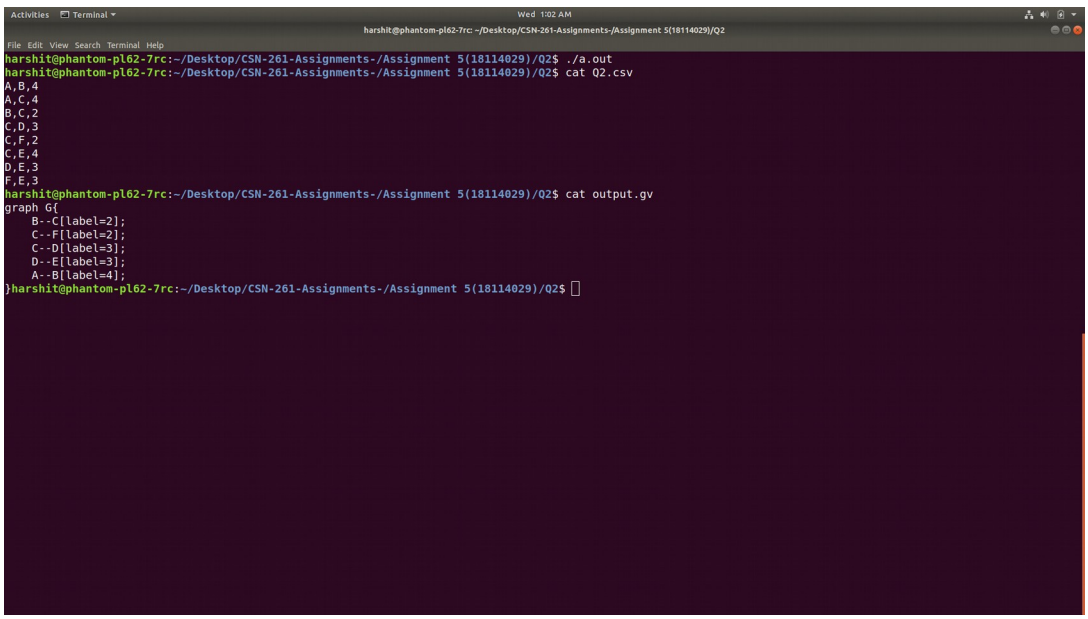


Question 2

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 graph G 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. For more details follow this link https://www.graphviz.org/doc/info/lang.html. Further use the "dot file" file to visualize the output graph in .pdf or .png file using Graphviz.

Algorithm: Kruskal MST
Data-structure: Hashmap(for graph), vector

Screenshots:



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. It is a greedy algorithm that finds a minimum spanning tree for a weighted undirected graph. This means it finds a subset of the edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. The algorithm generates the MST by adding one vertex at a time, starting from an arbitrary vertex. At each step the cheapest possible edge weight is chosen from the already selected vertex. These algorithms find the minimum spanning forest in a possibly disconnected graph; in contrast, the most basic form of Prim's algorithm only finds minimum spanning tree in connected graphs.

Algorithm: Prim’s MST
Data-structure: Fibonacci Heap