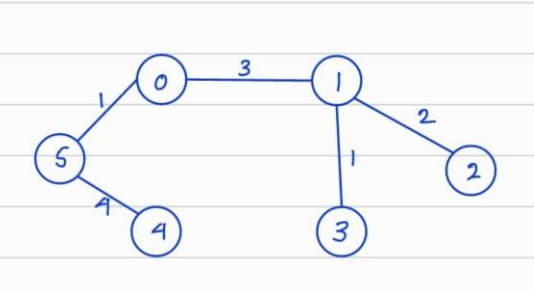
In Class 11

200356A

Q1.)

	0	**1**	**2**	**3**	**4**	**5**
0	0	3	0	0	0	1
1	3	0	2	1	10	0
2	0	2	0	3	0	5
3	0	1	3	0	5	0
4	0	10	0	5	0	4
5	1	0	5	0	4	0

Q2.)



Q3.)

```
● eegana\OneDrive - University of Moratuwa\Sem 4 Dulan\CS2023_data_structures_and_algorithms\InClass\LAB11 May23\" ; if ($?) { g++ lab11.cpp -o lab11 } ; if ($?)
  { .\lab11 }
 start    end     weight
 0        5       1
 0        1       3
 1        3       1
 1        2       2
 5        4       4
○ PS C:\Users\Dulan Lokugeegana\OneDrive - University of Moratuwa\Sem 4 Dulan\CS2023_data_structures_and_algorithms\InClass\LAB11 May23>
```

Q4.)

Yes.

When the weightage of the edges of a graph is unique it will result in a unique minimum spanning tree.

Q5.)

Time complexity of Prims algorithm is $O(V^2)$ and time complexity of Kruskal's algorithm is O(ElogV).

In Kruskal's algorithm we should first do merge sort and find the minimum weightage edge. For that the time complexity is O(ElogE). Then we connect the minimum weightage edge in such a way that there will be no cycles. Time complexity for the above-mentioned operation is O(ElogV).

When we implement prim's algorithm in normal way it will cost a time complexity of $O(V^2)$. But we can make an optimized code in time complexity of O(ElogV).

GitHub repository link
https://github.com/Dulan24/S4-CS2023-DSA-labs