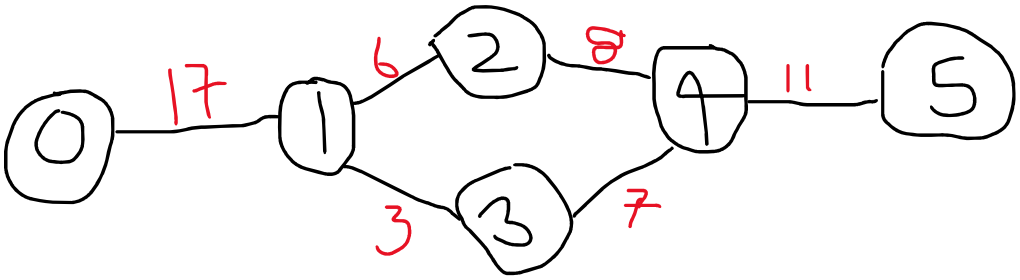
IMPLEMENTATION REPORT -ASSIGNMENT_2

Q1. Dijkstra's algorithm using an adjacency list. The following is the graph to be taken in account .In general , the time complexity of dijkstra's algorithm is vlog(e) and if the number of vertices == edges ==n , we get the time O(nlog(n)). A small example of sparse graph shown below.



A function called Dijkstra is made;

An array is made denoting the distances-

0	0
1	inf
2	Inf
3	Inf
4	Inf
5	inf

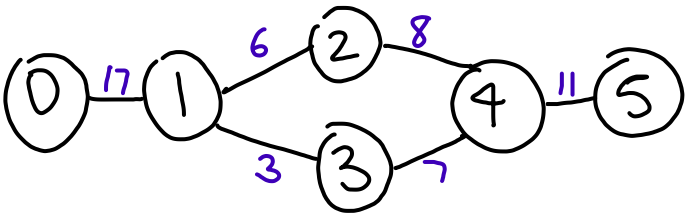
Then adjacency list representation

0	1
1	2,3
2	4
3	4
4	5
5	4

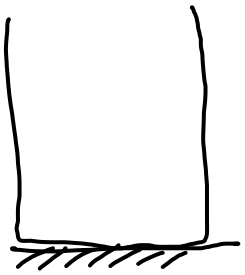
More like this format below:-

0	1	2	3	4
↓	↓	↓	↓	↓
(1,17)	(2,6)	(4,8)	(4,7)	(5,11)
	(3,3)			

The latter is the weight of the move.Now initialising the priority queue with min-heap , pushing the root node first.

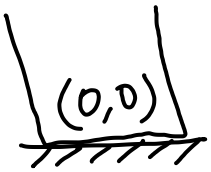


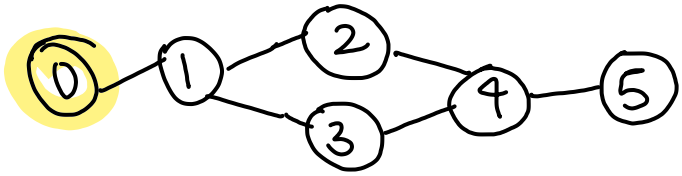
PQ

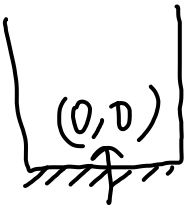


All not visited
Visited shown through

① Initially

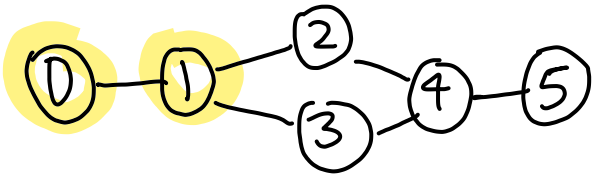


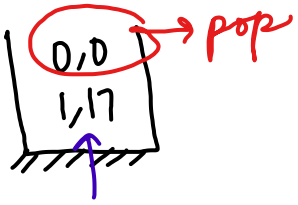




(1,17)

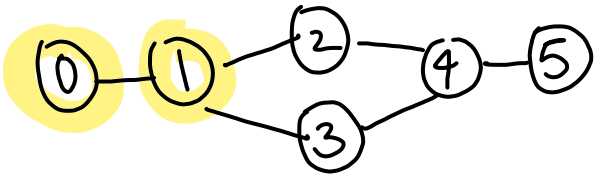
distance[1] = 17

② 

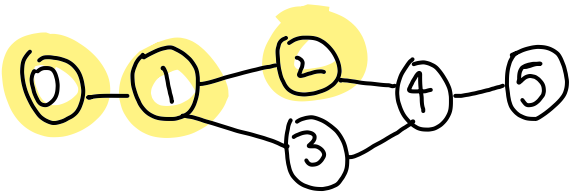


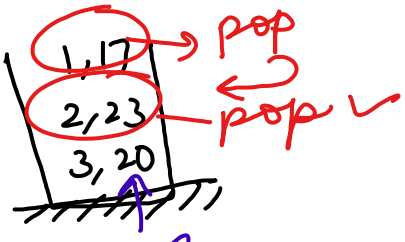
2,23

distance[2] = 17+6 = 23

(2.1) 

distance[3] = 17+3 = 20

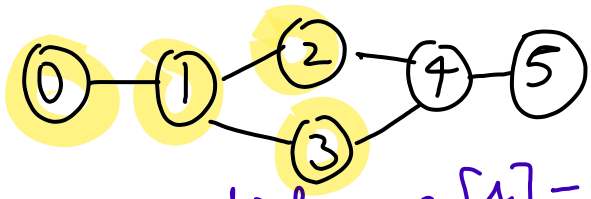
③ 

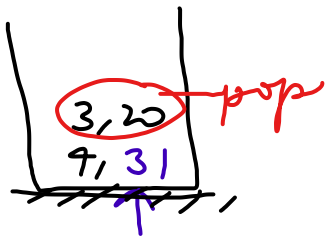


4,31

distance[4] = 23+8 = ~~28~~ 31

4



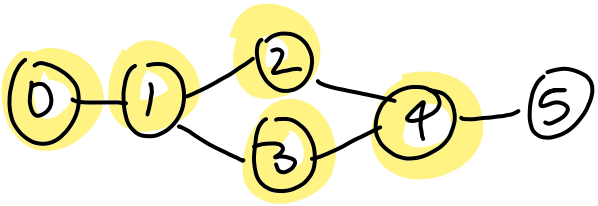


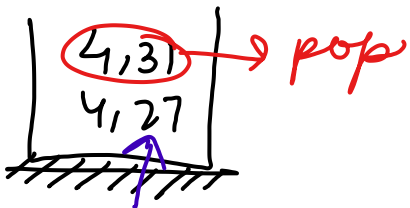
distance[4] = 20 + 7
= 27

(4,27)

UPDATED

5

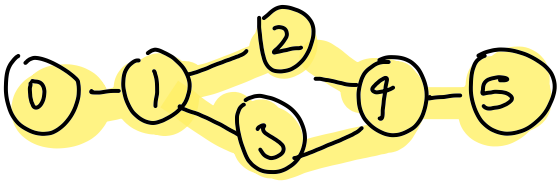


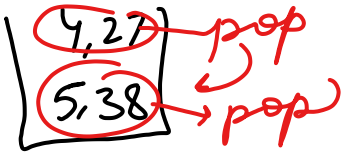


distance[5] = 27 + 11
= 38

(5,38)

6





no neat node
hence finished!