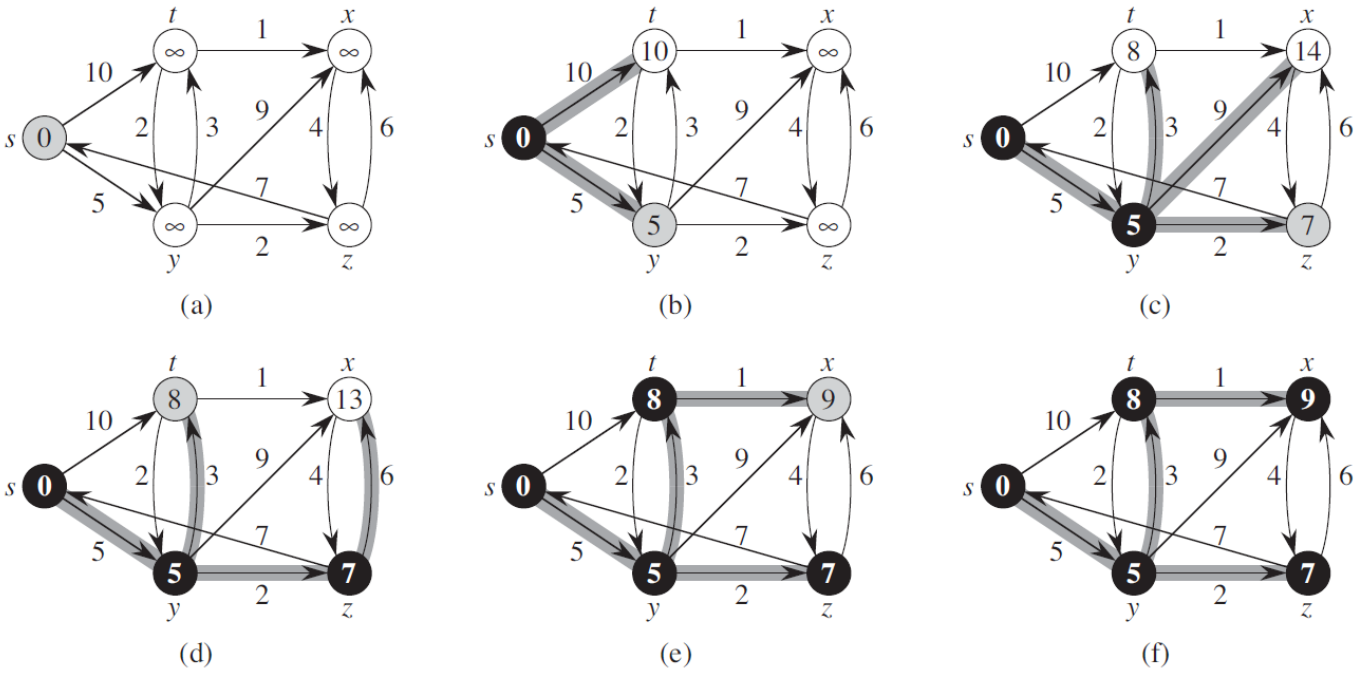
Offline 4 on Shortest Path Problems

In this assignment, you will implement the Dijkstra's algorithm to find out the shortest path from a source node to any of the other nodes.

How Dijkstra's Algorithm works:



(a) (b) (c)

(d) (e) (f)

Dijkstra's Algorithm:

```
DIJKSTRA(G, w, s)
    INITIALIZE-SINGLE-SOURCE(V, s)
    S ← ∅
    Q ← V[G]
    while Q ≠ ∅
        do u ← EXTRACT-MIN(Q)
            S ← S ∪ {u}
            for each vertex v ∈ Adj[u]
                do RELAX(u, v, w)
```

```
INITIALIZE-SINGLE-SOURCE(V, s)

    for each v ∈ V
        do d[v] ← ∞
            π[v] ← NIL
    d[s] ← 0
```

```
RELAX(u, v, w)
    if d[v] > d[u] + w(u, v)
        then d[v] ← d[u] + w(u, v)
            π[v] ← u
            also update the Priority Queue
                                    value
```

You must take input the graph from a .txt file and you must build an adjacency list (i.e. array of vectors). For minimum priority queue, you must use the priority_queue C++ stl library.

In the output, you will print all the shortest paths from the source node to any other nodes.

Sample Input and Output:

Sample Input	**Sample Output**

5 10
1 2 10
1 3 5
3 2 3
2 3 2
3 5 2
5 4 6
4 5 4
5 1 7
3 4 9
2 4 1
	Shortest path from 1 to 2 : -->1-->3-->2
	Shortest path from 1 to 3 : -->1-->3
	Shortest path from 1 to 4 : -->1-->3-->2-->4
	Shortest path from 1 to 5 : -->1-->3-->5

[Miss at your own risk]
[Zero tolerance for plagiarism]