Assignment 2: Single-Source Shortest Path

Due: 4th week (submission through moodle before sessional class)

Input: First line and second line of input file will contain the number of vertices, *n* and number of edges, *m* respectively followed by *m* lines each containing source, destination and weight of an edge of the directed graph.

For example:

```
5
8
1 3 6
1 4 3
2 1 3
3 4 2
4 2 1
4 3 1
5 2 4
5 4 2
```

Section B1
Implement Bellman-Ford algorithm for solving the single-source shortest-paths problem in the general case in which edge weights may be negative. The source vertex will be given as an input in console. You need to calculate shortest path estimate (v.d) and parent (v.π) for each vertex for the given source vertex. Your algorithm should run in time O(*VE*). Also detect whether the graph contains any negative weight cycle.

Output:
The Source? 5

V	V.d	V.π
1	6	2
2	3	4
3	3	4
4	2	5
5	0	nil

Negative Cycle? No

Section A1
Implement Bellman-Ford algorithm for solving the single-source shortest-paths problem in the general case in which edge weights may be negative. The source vertex will be given as an input in console. You need to calculate shortest path estimate (v.d) and parent (v.π) for each vertex for the given source

vertex. Your algorithm should run in time O(*VE*). Also detect whether the graph contains any negative weight cycle.

Output:
The Source? 5

V	V.d	V.π
1	6	2
2	3	4
3	3	4
4	2	5
5	0	nil

Negative Cycle? No

Section B2
Implement Bellman-Ford algorithm for solving the single-source shortest-paths problem in the general case in which edge weights may be negative. The source vertex will be given as an input in console. You need to calculate shortest path estimate (v.d) and parent (v.π) for each vertex for the given source vertex. Your algorithm should run in time O(*VE*). Also detect whether the graph contains any negative weight cycle.

Output:
The Source? 5

V	V.d	V.π
1	6	2
2	3	4
3	3	4
4	2	5
5	0	nil

Negative Cycle? No

Section A2
a) Implement Dijktra's algorithm for solving the single-source shortest-paths problem. The source vertex will be given as an input in console. You need to calculate shortest path estimate (v.d) and parent (v.π) for each vertex for the given source vertex. Use the heap you implemented during CSE204. Your algorithm should run in time O(*V*lg*V+E*).

Output:
The Source? 5

V	V.d	V.π
1	6	2
2	3	4
3	3	4
4	2	5
5	0	nil