Artificial Intelligence
(CS13217)

Lab Report

Name:	Wardha kanwal
Registration #:	CSU-S15-103
Lab Report #:	05
Dated:	18-05-2018
Submitted To:	Mr. Usman Ahmed

The University of Lahore, Islamabad Campus
Department of Computer Science & Information Technology

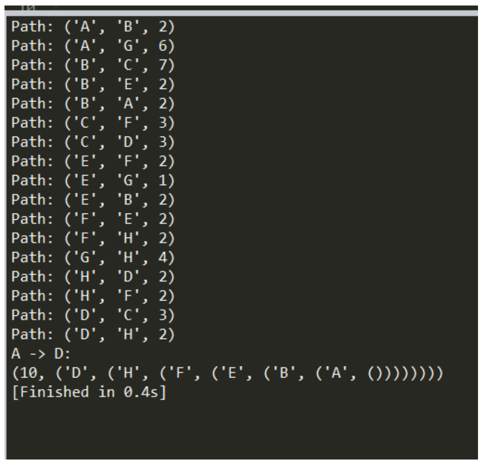
Experiment # 5
Dijkstra's Algorithm

Objective
To understand and implement the Dijkstra's problem.

Software Tool
1. Python
2. Sublime, version 3.0
3. Operating System, window 8.1

1 Theory

The Dijsktras algorithm compute the shortest paths in a graph from a single source node to all other nodes is common practice in industry and academia. The Dijsktras algorithm introduces an even faster queue design for the algorithm, find the path with the lowest cost between the vertex and every other vertex.
Dijsktras algorithm performance depends on the queue implementation.
1.Source node is initialized and can be indicated as filled circle.
2.Initial path cost to neighboring nodes (adjacent nodes) or link cost is computed and these nodes are relabeled considering source node.
3.Examine all adjacent nodes and find smallest label, make it permanent.
4.The smallest label is now working node, then Step 2 and Step 3 are repeated till the destination node is reached.
The Dijkstras algorithm is used for finding shortest path .It is used almost everywherefor example, the algorithm is used in the following areas:
1. Traffic Information Systems
2. Mapping (Google Maps or other mapping services or software)
3. Routing Systems



```
Path: ('A', 'B', 2)
Path: ('A', 'G', 6)
Path: ('B', 'C', 7)
Path: ('B', 'E', 2)
Path: ('B', 'A', 2)
Path: ('C', 'F', 3)
Path: ('C', 'D', 3)
Path: ('E', 'F', 2)
Path: ('E', 'G', 1)
Path: ('E', 'B', 2)
Path: ('F', 'E', 2)
Path: ('F', 'H', 2)
Path: ('G', 'H', 4)
Path: ('H', 'D', 2)
Path: ('H', 'F', 2)
Path: ('D', 'C', 3)
Path: ('D', 'H', 2)
A -> D:
(10, ('D', ('H', ('F', ('E', ('B', ('A', ())))))))
[Finished in 0.4s]
```

Figure 1: Time Independent Feature Set

2 Task

2.1 Procedure: Task 1

1.Source node is initialized and can be indicated as filled circle.
2.Initial path cost to neighboring nodes (adjacent nodes) or link cost is computed and these nodes are relabeled considering source node.
3.Examine all adjacent nodes and find smallest label, make it permanent.
4.The smallest label is now working node, then Step 2 and Step 3 are repeated till the destination node is reached.

2.2 Procedure: Task 2

```
from collections import defaultdict
from heapq import *

def dijkstra(edges, f, t):
    g = defaultdict(list)
    for l,r,c in edges:
        g[l].append((c,r))
    q, seen = [(0,f,())], set()
    while q:
        (cost,v1,path) = heappop(q)
        if v1 not in seen:
            seen.add(v1)
            path = (v1, path)
            if v1 == t: return (cost, path)

            for c, v2 in g.get(v1, ()):
                if v2 not in seen:
                    heappush(q, (cost+c, v2, path))

    return float("inf")
if __name__ == "__main__":
    edges = [
        ("A", "B", 2),
        ("A", "G", 6),
        ("B", "C", 7),
        ("B", "E", 2),
        ("B", "A", 2),
        ("C", "F", 3),
        ("C", "D", 3),
        ("E", "F", 2),
        ("E", "G", 1),
        ("E", "B", 2),
        ("F", "E", 2),
        ("F", "H", 2),
        ("G", "H", 4),
        ("H", "D", 2),
        ("H", "F", 2),
        ("D", "C", 3),
        ("D", "H", 2),
    ]

    for x in range (len(edges)):
        print "Path:",edges[x]
    print "A -> D:"
    print dijkstra(edges, "A", "D")
```

3 Conclusion

In conclusion, the Dijkstras algorithm is used for finding shortest path .It is used almost everywherefor example, the algorithm is used in the following areas:
1. Traffic Information Systems
2. Mapping (Google Maps or other mapping services or software)
3. Routing Systems