CS671 Artificial Intelligence

Programming 1

Shuowen Wei

Math Department, weis11@wfu.edu

The map file, saved as **map.txt**

Oradea Zerind 71
Zerind Arad 75
Arad Sibiu 140
Oradea Sibiu 151
Arad Timisoara 118
Timisoara Lugoj 111
Lugoj Mehadia 70
Mehadia Dobreta 75
Dobreta Craiova 120
Craiova Rimnicu 146
Rimnicu Sibiu 80
Sibiu Fagaras 99
Fagaras Bucharest 211
Craiova Pitesti 138
Pitesti Rimnicu 97
Pitesti Bucharest 101
Bucharest Giurgiu 90
Bucharest Urziceni 85
Urziceni Hirsova 98
Hirsova Eforie 86
Urziceni Vaslui 142
Vaslui Lasi 92
Lasi Neamt 87

- **Sample output 1:**

Input:

map, initial city=’Arad’, destination city=’Bucharest’

➢ Algorithm 1: breadth-first Search

```
clc;clear
initial='Arad';
dstn_city='Bucharest';
map=importdata('map.txt');

[total_cost] = bfs(map,initial,dstn_city)
```

Output:

```
    'Arad'
    'Sibiu'
    'Fagaras'
    'Bucharest'
total_cost =
   450
```

➢ Algorithm 2: depth-first Search

```
clc;clear
initial='Arad';
dstn_city='Bucharest';
map=importdata('map.txt');

[total_cost] = dfs(map,initial,dstn_city)
```

Output:

```
    'Arad'
    'Timisoara'
    'Lugoj'
    'Mehadia'
    'Dobreta'
    'Craiova'
    'Pitesti'
    'Rimnicu'
    'Sibiu'
    'Fagaras'
    'Bucharest'
total_cost =
        1119
```

➢ Algorithm 3: iterative deepening depth-first Search

```
clc;clear
initial='Arad';
dstn_city='Bucharest';
map=importdata('map.txt');

[total_cost] = iterative_dfs(map,initial,dstn_city)
```

Output:

```
    'Arad'
    'Sibiu'
    'Fagaras'
    'Bucharest'
total_cost =
   450
```

- **Sample output 2:**

Input:

map, initial city=’Sibiu’, destination city=’Eforie’

➢ Algorithm 1: breadth-first Search

```
clc;clear
initial='Sibiu';
dstn_city='Eforie';
map=importdata('map.txt');

[total_cost] = bfs(map,initial,dstn_city)
```

Output:

```
    'Sibiu'
    'Fagaras'
    'Bucharest'
    'Urziceni'
    'Hirsova'
    'Eforie'
total_cost =
   579
```

➢ Algorithm 2: depth-first Search

```
clc;clear
initial='Sibiu';
dstn_city='Eforie';
map=importdata('map.txt');

[total_cost] = dfs(map,initial,dstn_city)
```

Output:

```
    'Sibiu'
    'Arad'
    'Timisoara'
    'Lugoj'
    'Mehadia'
    'Dobreta'
    'Craiova'
    'Rimnicu'
    'Pitesti'
    'Bucharest'
    'Urziceni'
    'Hirsova'
    'Eforie'
total_cost =
        1247
```

- **Algorithm 3: iterative deepening depth-first Search**

```
clc;clear
initial='Sibiu';
dstn_city='Eforie';
map=importdata('map.txt');

[total_cost] = iterative_dfs(map,initial,dstn_city)
```

Output:

```
    'Sibiu'
    'Fagaras'
    'Bucharest'
    'Urziceni'
    'Hirsova'
    'Eforie'
total_cost =
   579
```

- **Sample output 3:**

Input:

map, initial city=’Dobreta’, destination city=’ Fagaras’

➢ Algorithm 1: breadth-first Search

```
clc;clear
initial='Dobreta';
dstn_city='Fagaras';
map=importdata('map.txt');

[total_cost] = bfs(map,initial,dstn_city)
```

Output:

```
    'Dobreta'
    'Craiova'
    'Rimnicu'
    'Sibiu'
    'Fagaras'
total_cost =
   445
```

➢ Algorithm 2: depth-first Search

```
clc;clear
initial='Dobreta';
dstn_city='Fagaras';
map=importdata('map.txt');

[total_cost] = dfs(map,initial,dstn_city)
```

Output:

```
    'Dobreta'
    'Mehadia'
    'Lugoj'
    'Timisoara'
    'Arad'
    'Sibiu'
    'Rimnicu'
    'Craiova'
    'Pitesti'
    'Bucharest'
    'Fagaras'
total_cost =
        1190
```

➢ Algorithm 3: iterative deepening depth-first Search

```
clc;clear
initial='Dobreta';
dstn_city='Fagaras';
```

```
map=importdata('map.txt');

[total_cost] = iterative_dfs(map,initial,dstn_city)
```

Output:

```
    'Dobreta'
    'Craiova'
    'Rimnicu'
    'Sibiu'
    'Fagaras'
total_cost =
   445
```