

SCREENSHOTS OF OUTPUTS

DIJKSTRA'S ALGORITHM - DIRECTED

```
prachi@Prachis-MacBook-Air DSALG0PROJ2 % /usr/bin/python3 /Users/prachi/Downloads/DSALG0PROJ2/source_Code.py
-----Shortest Path Dijkstra Algorithm-----

Shortest path for the Graph 0

Number of Vertices: 6
Number of Edges: 10
The Graph is DIRECTED
Source vertex: A

Cost of shortest path from the node A -> B is 1 and path is A B
Cost of shortest path from the node A -> C is 2 and path is A C
Cost of shortest path from the node A -> D is 3 and path is A C D
Cost of shortest path from the node A -> E is 3 and path is A B E
Cost of shortest path from the node A -> F is 6 and path is A C D F

Runtime for Dijkstras Algorithm: 1 Seconds

Shortest path for the Graph 1

Number of Vertices: 11
Number of Edges: 21
The Graph is DIRECTED
Source vertex: A

Cost of shortest path from the node A -> B is 4 and path is A B
Cost of shortest path from the node A -> C is 15 and path is A B C
Cost of shortest path from the node A -> E is 15 and path is A B D E
Cost of shortest path from the node A -> D is 13 and path is A B D
Cost of shortest path from the node A -> F is 19 and path is A B D F
Cost of shortest path from the node A -> G is 22 and path is A B D E G
Cost of shortest path from the node A -> H is 19 and path is A B D E H
Cost of shortest path from the node A -> I is 29 and path is A B D E H I
Cost of shortest path from the node A -> K is 30 and path is A B D E H K
Cost of shortest path from the node A -> J is 31 and path is A B D E H I J

Runtime for Dijkstras Algorithm: 1 Seconds
```

DIJKSTRA'S ALGORITHM UNDIRECTED

Shortest path for the Graph 5

Number of Vertices: 10

Number of Edges: 11

The Graph is UNDIRECTED

Source vertex: B

Cost of shortest path from the node B -> A is	3 and path is	B A
Cost of shortest path from the node B -> D is	4 and path is	B A D
Cost of shortest path from the node B -> E is	8 and path is	B A E
Cost of shortest path from the node B -> C is	4 and path is	B C
Cost of shortest path from the node B -> F is	14 and path is	B A E F
Cost of shortest path from the node B -> G is	21 and path is	B A E F G
Cost of shortest path from the node B -> H is	23 and path is	B A E F H
Cost of shortest path from the node B -> I is	29 and path is	B A E F G I
Cost of shortest path from the node B -> J is	34 and path is	B A E F H J

Runtime for Dijkstras Algorithm: 1 Seconds

Shortest path for the Graph 6

Number of Vertices: 13

Number of Edges: 25

The Graph is UNDIRECTED

Source vertex: A

Cost of shortest path from the node A -> D is	1 and path is	A D
Cost of shortest path from the node A -> B is	4 and path is	A B
Cost of shortest path from the node A -> E is	3 and path is	A E
Cost of shortest path from the node A -> H is	25 and path is	A D H
Cost of shortest path from the node A -> I is	24 and path is	A E I
Cost of shortest path from the node A -> F is	10 and path is	A B F
Cost of shortest path from the node A -> C is	11 and path is	A B C
Cost of shortest path from the node A -> G is	18 and path is	A B F G
Cost of shortest path from the node A -> J is	28 and path is	A B F J
Cost of shortest path from the node A -> K is	20 and path is	A B C K
Cost of shortest path from the node A -> L is	32 and path is	A B F G L
Cost of shortest path from the node A -> M is	31 and path is	A B F G M

Runtime for Dijkstras Algorithm: 1 Seconds

KRUSKAL'S ALGORITHM

-----Minimum Spanning Tree (Kruskal Algorithm)-----

Minimum Spanning Tree for graph 5

Edge Selected	Weight
---------------	--------

A -> D	1
D -> C	2
A -> B	3
A -> E	5
F -> E	6
F -> G	7
G -> I	8
F -> H	9
H -> J	11

The total Cost for Minimum Spanning Tree is 52

Runtime for Kruskal Algorithm: 1 Seconds

Minimum Spanning Tree for graph 6

Edge Selected	Weight
---------------	--------

A -> D	1
D -> E	2
A -> B	4
B -> F	6
B -> C	7
F -> G	8
C -> K	9
K -> M	12
G -> L	14
J -> L	16
I -> J	19
H -> I	22

The total Cost for Minimum Spanning Tree is 120

Runtime for Kruskal Algorithm: 1 Seconds

STRONGLY CONNECTED GRAPHS

```
-----Strongly Connected Components-----  
Strongly connected components for graph0  
No.of vertices:  6  
F E D C B A  
Runtime for calculating SCC: 1 Seconds  
  
Strongly connected components for graph1  
No.of vertices:  11  
K J I F H G E D C B A  
Runtime for calculating SCC: 1 Seconds  
  
Strongly connected components for graph2  
No.of vertices:  12  
K J I L H B F G E D C A  
Runtime for calculating SCC: 1 Seconds  
  
Strongly connected components for graph3  
No.of vertices:  11  
F N K T G E D C A S B  
Runtime for calculating SCC: 1 Seconds  
  
Strongly connected components for graph4  
No.of vertices:  13  
M L K I C D H A B F E J G  
Runtime for calculating SCC: 1 Seconds  
  
Strongly connected components for graph5  
No.of vertices:  10  
C B D E A I G J H F  
Runtime for calculating SCC: 1 Seconds  
  
Strongly connected components for graph6  
No.of vertices:  13  
L J I H M G F E D K C B A  
Runtime for calculating SCC: 1 Seconds
```