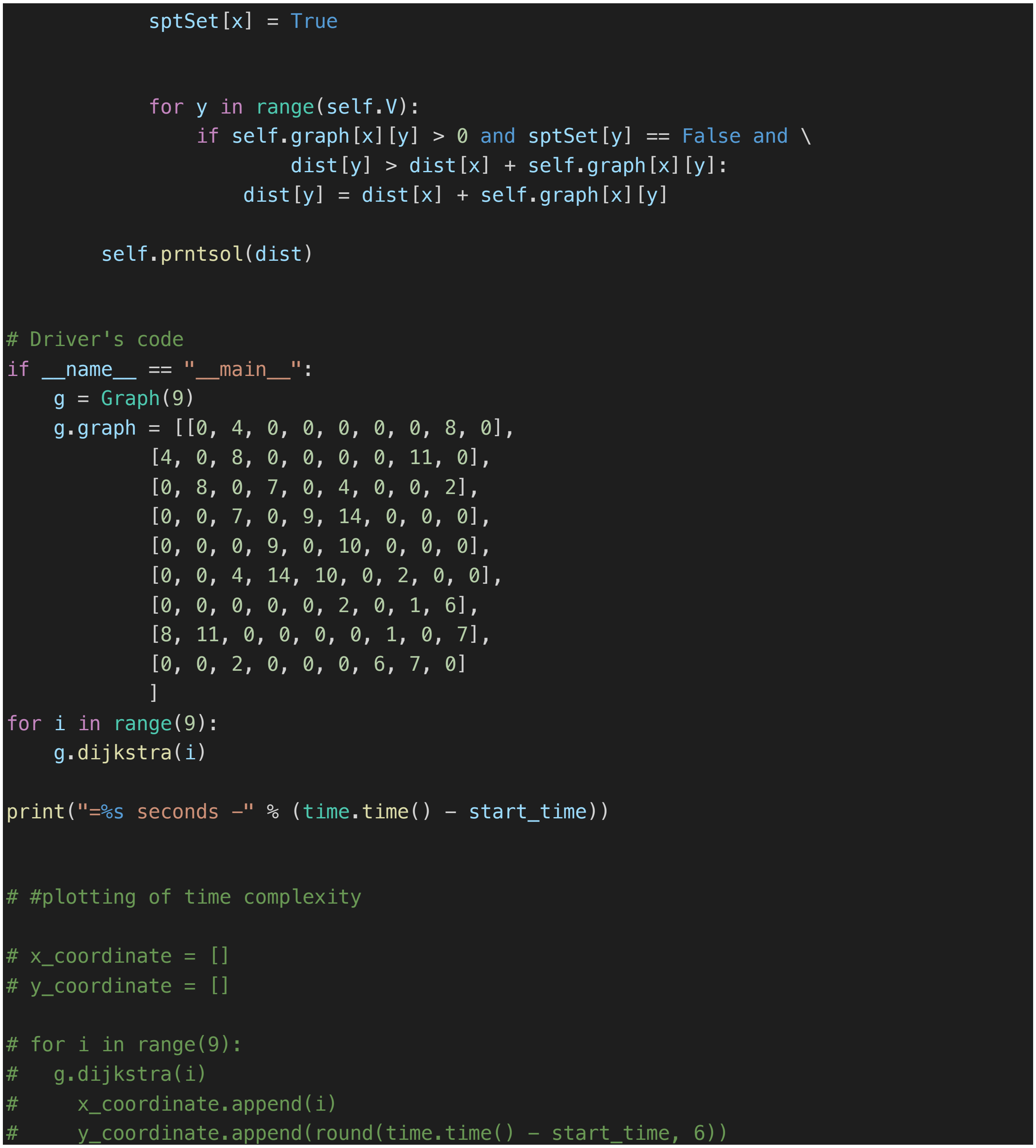
Quiz 6

Source code:

```
import sys
from turtle import clear
import time
import matplotlib.pyplot as plt

start_time=time.time()

class Graph():
    def __init__(self, vertices):
        self.V = vertices
        self.graph = [[0 for column in range(vertices)]
                      for row in range(vertices)]

    def prntsol(self, dist):
        print("Vertex \tDistance from Source")
        for node in range(self.V):
            print(node, "\t", dist[node])


    def mindist(self, dist, sptSet):

        min = sys.maxsize


        for u in range(self.V):
            if dist[u] < min and sptSet[u] == False:
                min = dist[u]
                min_index = u

        return min_index


    def dijkstra(self, src):

        dist = [sys.maxsize] * self.V
        dist[src] = 0
        sptSet = [False] * self.V

        for cout in range(self.V):


            x = self.mindist(dist, sptSet)
```



```
            sptSet[x] = True


            for y in range(self.V):
                if self.graph[x][y] > 0 and sptSet[y] == False and \
                        dist[y] > dist[x] + self.graph[x][y]:
                    dist[y] = dist[x] + self.graph[x][y]

        self.prntsol(dist)


# Driver's code
if __name__ == "__main__":
    g = Graph(9)
    g.graph = [[0, 4, 0, 0, 0, 0, 0, 8, 0],
            [4, 0, 8, 0, 0, 0, 0, 11, 0],
            [0, 8, 0, 7, 0, 4, 0, 0, 2],
            [0, 0, 7, 0, 9, 14, 0, 0, 0],
            [0, 0, 0, 9, 0, 10, 0, 0, 0],
            [0, 0, 4, 14, 10, 0, 2, 0, 0],
            [0, 0, 0, 0, 0, 2, 0, 1, 6],
            [8, 11, 0, 0, 0, 0, 1, 0, 7],
            [0, 0, 2, 0, 0, 0, 6, 7, 0]
            ]
for i in range(9):
    g.dijkstra(i)

print("=%s seconds -" % (time.time() - start_time))


# #plotting of time complexity

# x_coordinate = []
# y_coordinate = []

# for i in range(9):
#   g.dijkstra(i)
#     x_coordinate.append(i)
#     y_coordinate.append(round(time.time() - start_time, 6))
```

Output:

```
harshavaidhyam@Harshas-MacBook-Pro quiz 6 %  cd /Users/harshavaidhyam/Desktop/Pitt\
term-1/Algo\ Design/quiz\ 6 ; /usr/bin/env /usr/local/bin/python3 /
```

```
Users/harshavaidhyam/.vscode/extensions/ms-python.python-
2022.16.0/pythonFiles/lib/python/debugpy/adapter/../../debugpy/launcher 52708 --
/Users/harsha
vaidhyam/Desktop/Pitt\ term-1/Algo\ Design/quiz\ 6/test3.py

0-> 1(4) -> 7(8)
1 -> 0(4) -> 2(8) -> 7(11)
7 -> 0(8) -> 1(11) -> 6(1) -> 8(7)
2 -> 1(8) -> 3(7) -> 8(2) -> 5(4)
3 -> 2(7) -> 4(9) -> 5(14)
8 -> 2(2) -> 6(6) -> 7(7)
5 -> 2(4) -> 3(14) -> 4(10) -> 6(2)
4 -> 3(9) -> 5(10)
6 -> 5(2) -> 7(1) -> 8(6)
Distance from node: 0
Node 0 has distance: 0
Node 1 has distance: 4
Node 2 has distance: 12
Node 3 has distance: 19
Node 4 has distance: 21
Node 5 has distance: 11
Node 6 has distance: 9
Node 7 has distance: 8
Node 8 has distance: 14
Traceback (most recent call last):
0 -> 1(4) -> 7(8)
1 -> 0(4) -> 2(8) -> 7(11)
7 -> 0(8) -> 1(11) -> 6(1) -> 8(7)
2 -> 1(8) -> 3(7) -> 8(2) -> 5(4)
3 -> 2(7) -> 4(9) -> 5(14)
8 -> 2(2) -> 6(6) -> 7(7)
5 -> 2(4) -> 3(14) -> 4(10) -> 6(2)
4 -> 3(9) -> 5(10)
6 -> 5(2) -> 7(1) -> 8(6)
Distance from node: 0
Node 0 has distance: 0
Node 1 has distance: 4
Node 2 has distance: 12
0 -> 1(4) -> 7(8)
1 -> 0(4) -> 2(8) -> 7(11)
7 -> 0(8) -> 1(11) -> 6(1) -> 8(7)
2 -> 1(8) -> 3(7) -> 8(2) -> 5(4)
3 -> 2(7) -> 4(9) -> 5(14)
8 -> 2(2) -> 6(6) -> 7(7)
```

```
5 -> 2(4) -> 3(14) -> 4(10) -> 6(2)
4 -> 3(9) -> 5(10)
6 -> 5(2) -> 7(1) -> 8(6)
Distance from node: 0
Node 0 has distance: 0
Node 1 has distance: 4
Node 2 has distance: 12
Node 3 has distance: 19
Node 4 has distance: 21
Node 5 has distance: 11
Node 6 has distance: 9
Node 7 has distance: 8
Node 8 has distance: 14
```

```
0 -> 1(4) -> 7(8)
1 -> 0(4) -> 2(8) -> 7(11)
7 -> 0(8) -> 1(11) -> 6(1) -> 8(7)
2 -> 1(8) -> 3(7) -> 8(2) -> 5(4)
3 -> 2(7) -> 4(9) -> 5(14)
8 -> 2(2) -> 6(6) -> 7(7)
5 -> 2(4) -> 3(14) -> 4(10) -> 6(2)
4 -> 3(9) -> 5(10)
6 -> 5(2) -> 7(1) -> 8(6)
Distance from node: 0
Node 0 has distance: 0
Node 1 has distance: 4
Node 2 has distance: 12
```

```
Vertex  Distance from Source
0       0
1       4
2       12
3       19
4       21
5       11
6       9
7       8
8       14
Vertex  Distance from Source
0       4
1       0
2       8
3       15
4       22
```

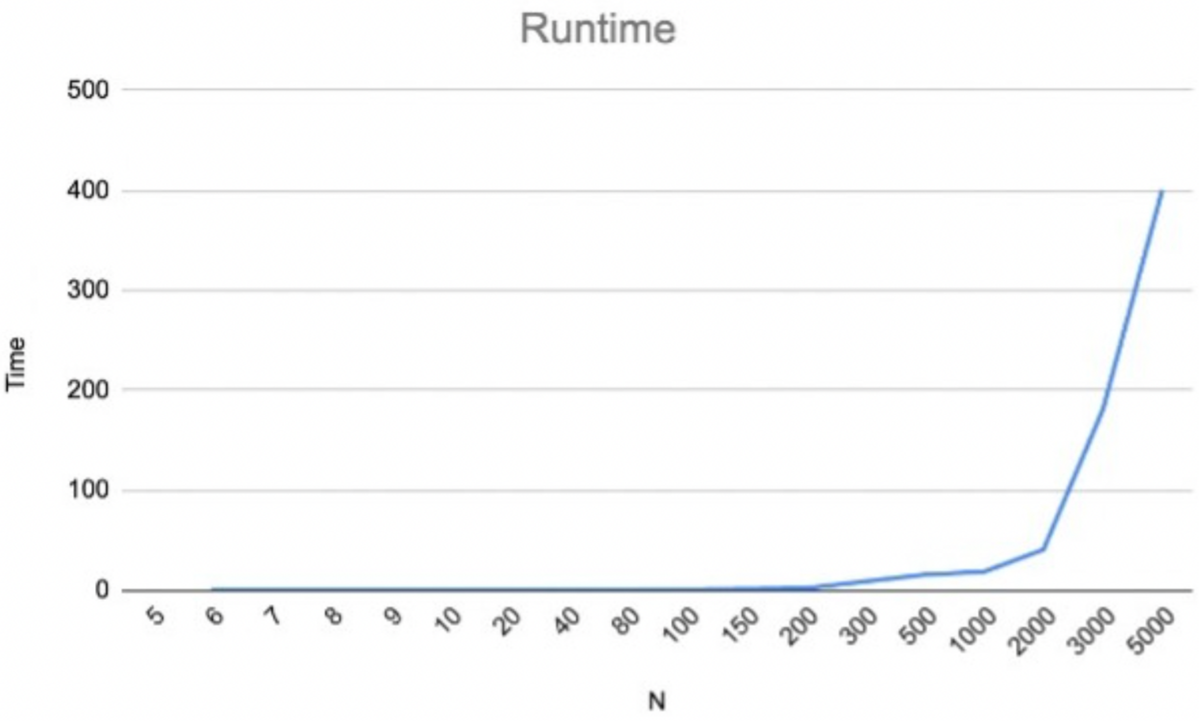
```
5	12
6	12
7	11
8	10
Vertex  Distance from Source
0	12
1	8
2	0
3	7
4	14
5	4
6	6
7	7
8	2
Vertex  Distance from Source
0	19
1	15
2	7
3	0
4	9
5	11
6	13
7	14
8	9
Vertex  Distance from Source
0	21
1	22
2	14
3	9
4	0
5	10
6	12
7	13
8	16
Vertex  Distance from Source
0	11
1	12
2	4
3	11
4	10
5	0
6	2
7	3
8	6
```

```
Vertex  Distance from Source
0       9
1       12
2       6
3       13
4       12
5       2
6       0
7       1
8       6
Vertex  Distance from Source
0       8
1       11
2       7
3       14
4       13
5       3
6       1
7       0
8       7
Vertex  Distance from Source
0       14
1       10
2       2
3       9
4       16
5       6
6       6
7       7
8       0
=0.0003509521484375 seconds –
```

Time complexity:

Theoretical: O(n2)

Experimental:
Since there are 3 nested for loops(first for when calling dijkstras function, and 2 for loops inside that function) because we are calculating shortest loop for each node, the experimental time complexity is:
O(n3)


Runtime
500
400
300
200
100
0
Time
5
6
7
8
9
10
20
40
80
100
150
200
300
500
1000
2000
3000
5000
N