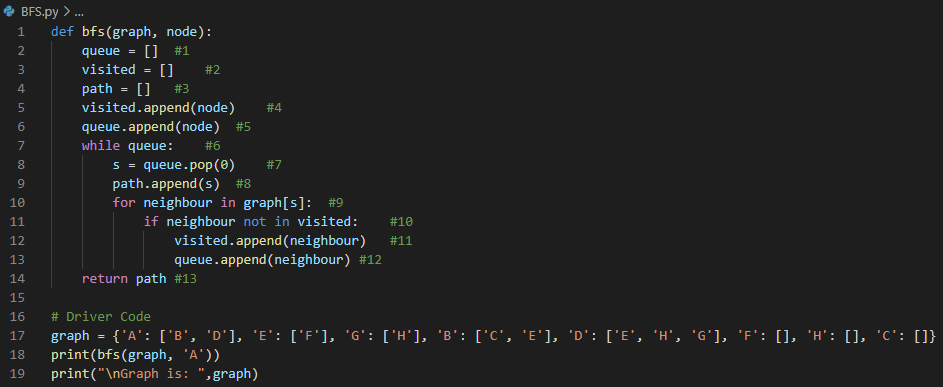
TALHA AHMED

UIT  18B-024-SE

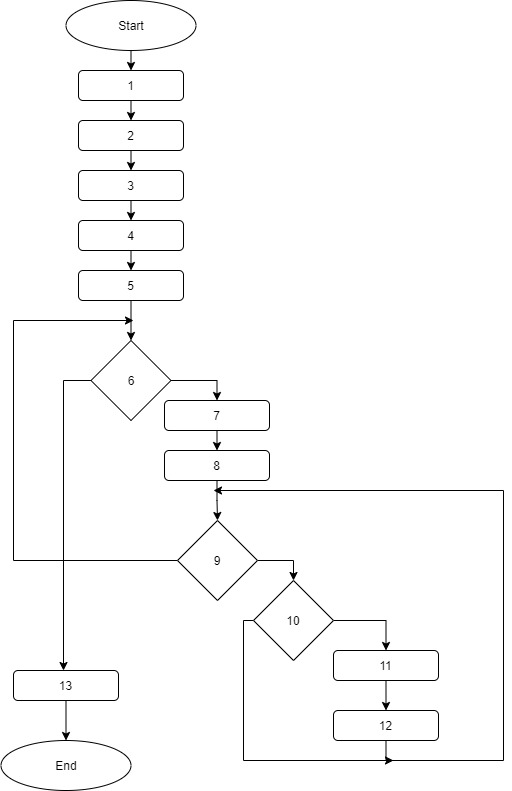
Whitebox Testing bfs-algorithm

**Code:**

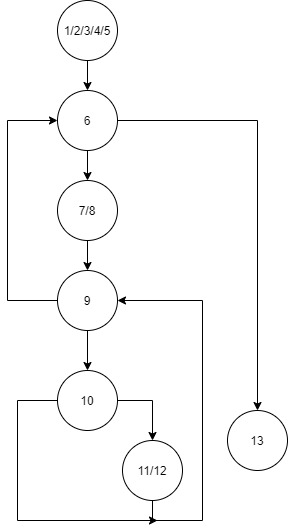
**Scenario:**

Implementation of BFS algorithm is that there is a graph and a first node. We provide node to the algorithm which is used as starting point to traverse the graph in breadth first order.

**Flowchart:**



**Flowgraph:**



**Cyclomatic Complexity:**

Regions = circular regions + 1 = 4

E – N + 2 = 9–7+2 = 4

Predicate Nodes + 1 = 4

**Path:**

1 2 3 4 5 6 13

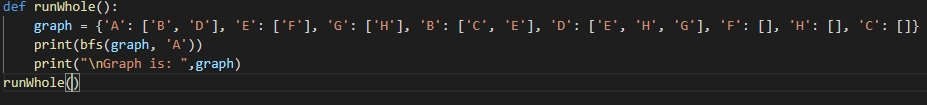
1 2 3 4 5 6 7 8 9 6 13

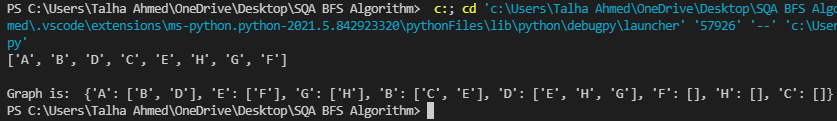
1 2 3 4 5 6 7 8 9 10 11 12 9 6 13

1 2 3 4 5 6 7 8 9 10 9 6 13

**TESTING**

**Running Whole Code:**

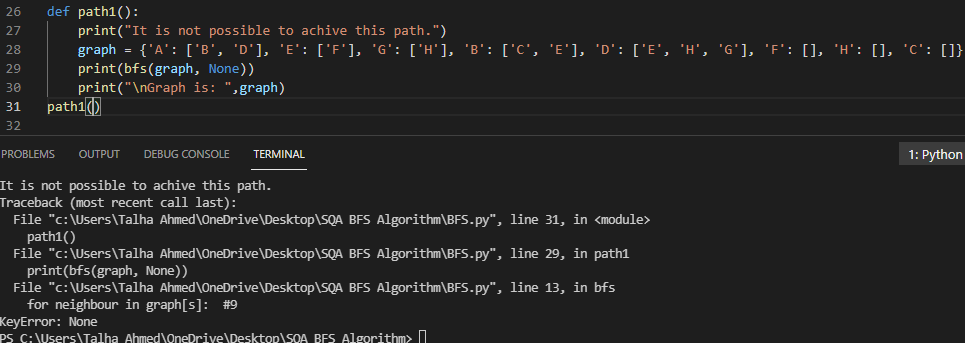




**Feedback:**

Running whole code smoothly by providing correct parameters.

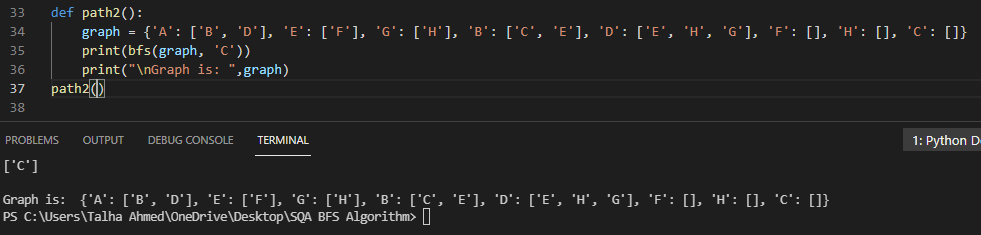
**Testing Path 1:**



**Feedback:**

It is not possible to achieve this path.

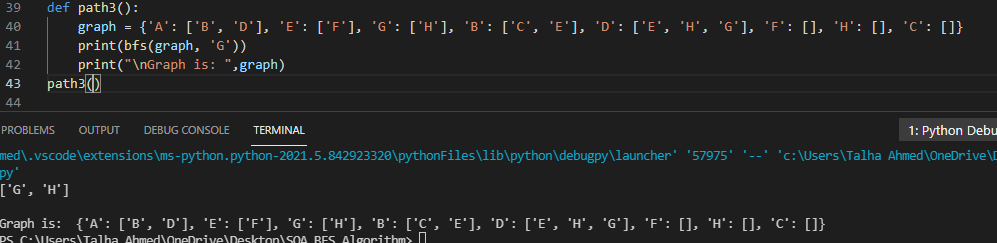
**Testing Path 2:**



**Feedback:**

Path 2 achieved successfully.

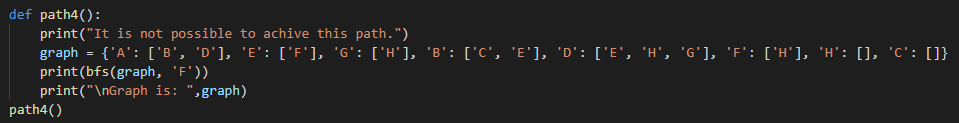
**Testing Path 3:**



**Feedback:**

Path 3 achieved successfully.

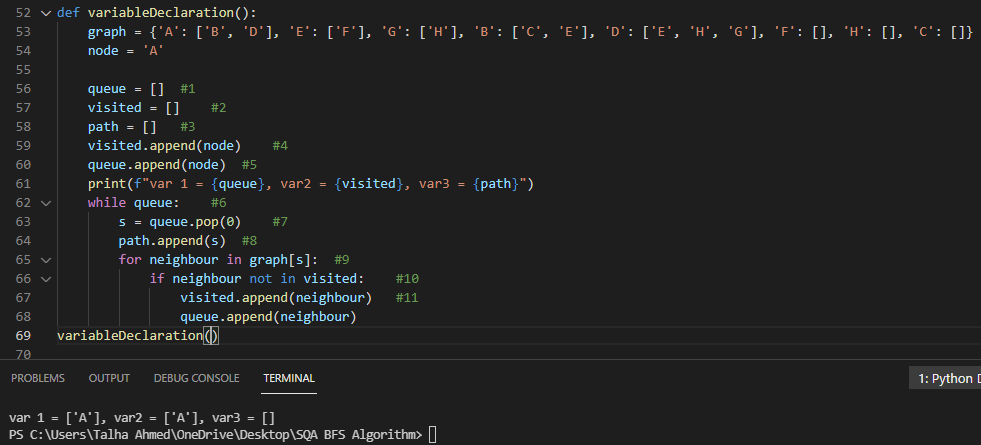
**Testing Path 4:**



**Feedback:**

It is not possible to achieve this path.

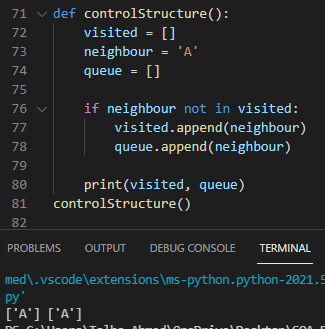
**Variable Declaration:**



**Feedback:**

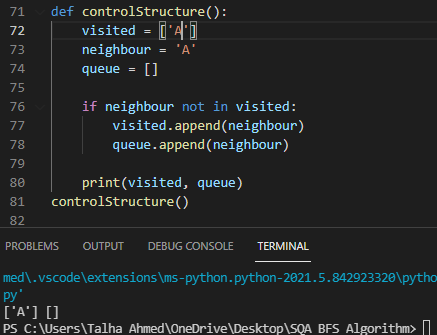
It is noticed that all variable which is declared are printed on console successfully.

**Control Structure:**



**Feedback:**

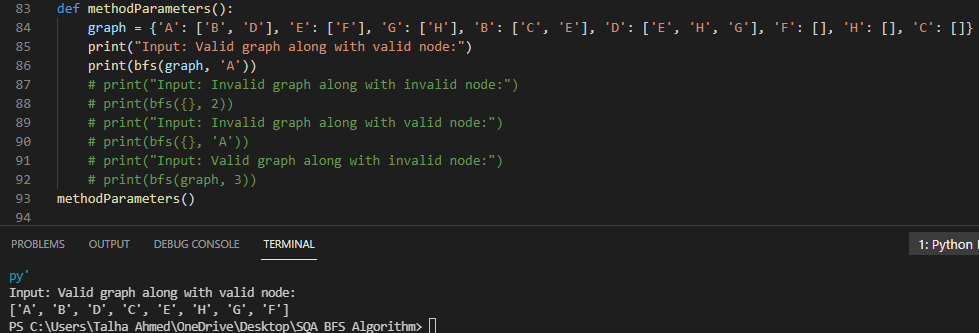
Boolean is tested here in which if condition is True and the queue and visited list is appended which is verified by printed in console.



**Feedback:**

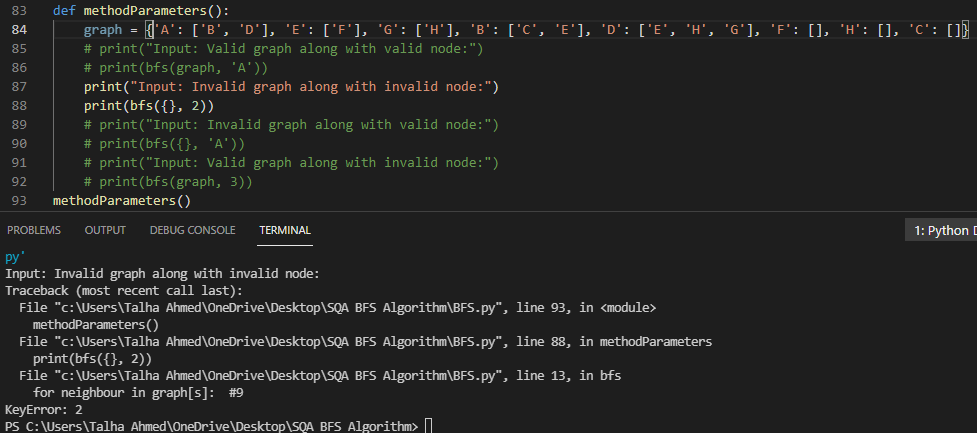
Boolean is tested here in which if condition is False and the queue and visited list is not appended which is verified by printing in console.

**Method Parameters:**



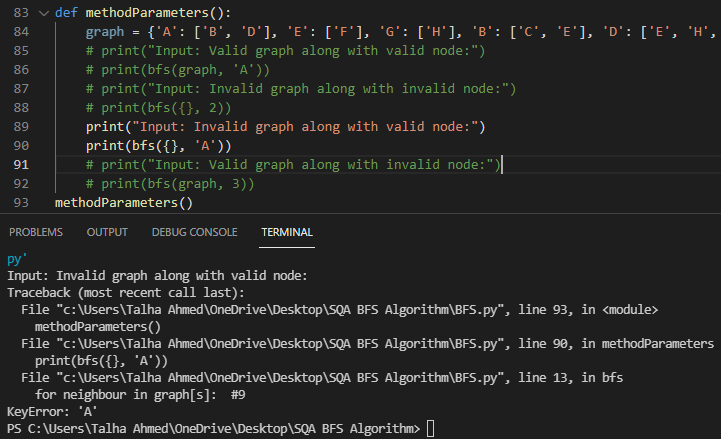
**Feedback:**

Input Valid graph along with valid node code runs successfully.



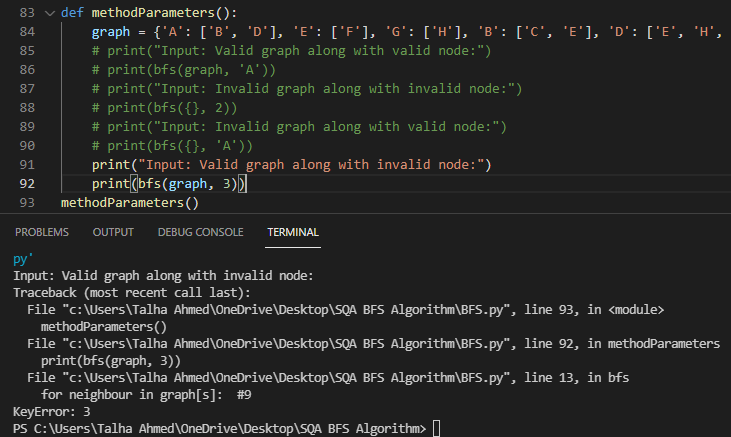
**Feedback:**

Input invalid graph along with invalid node code gives key error.



**Feedback:**

Input invalid graph along with valid node code gives key error.

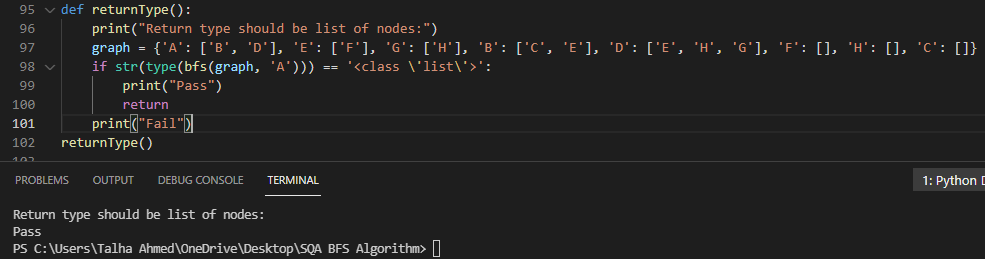


**Feedback:**

Input valid graph along with invalid node code gives key error.

So, it is concluded that code only runs on the valid graph and valid node.

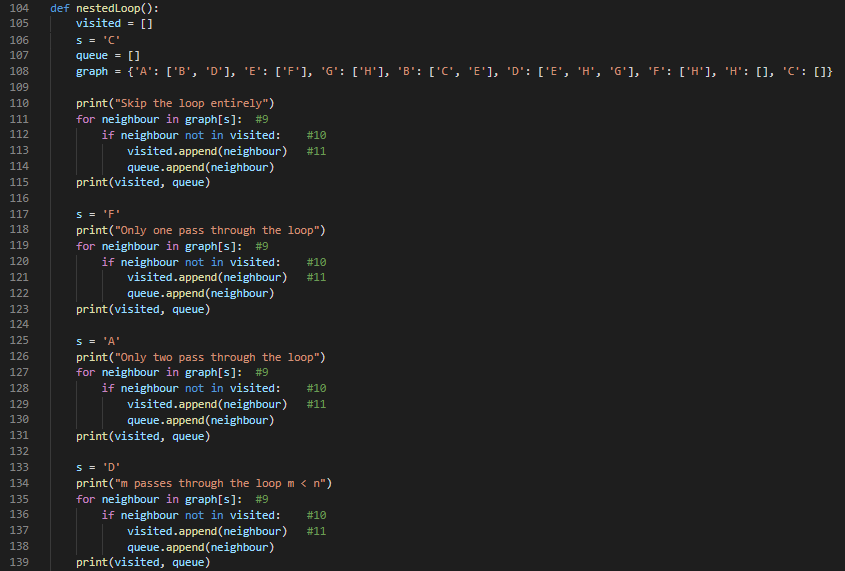
**Return Type:**

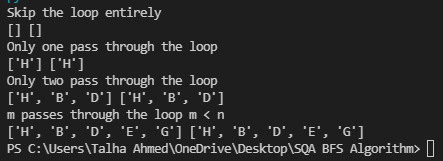


**Feedback:**

Return type should be the list of nodes and the test passes which shows that the return type is valid which is list of nodes.

**Nested Loop:**

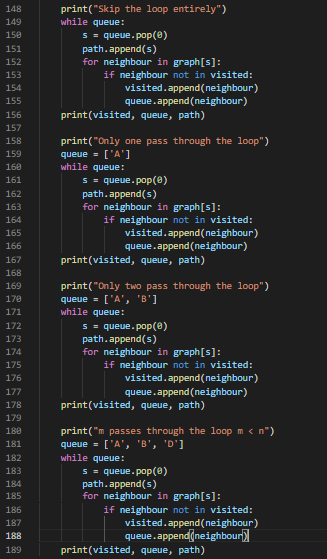


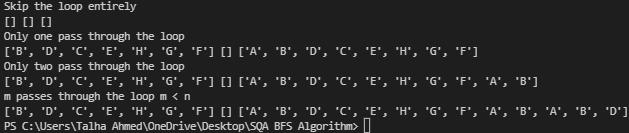


**Feedback:**

1. First skip the loop entirely and it passes the test that code runs smoothly without loop.
2. Only one pass through the loop and loop fulfills the expected output.
3. Only two passes through the loop and loop fulfills the expected output.
4. M passes through the loop which is 3 and M < N and loop fulfills the expected output.

**Outer Loop:**





**Feedback:**

1. First skip the loop entirely and it passes the test that code runs smoothly without loop.
2. Only one pass through the loop and loop fulfills the expected output.
3. Only two passes through the loop and loop fulfills the expected output.
4. M passes through the loop which is 3 and M < N and loop fulfills the expected output.