DEPARTMENT OF
COMPUTER SCIENCE & ENGINEERING
Discover. Learn. Empower.

Experiment1.3

Student Name: Jaiswal Mukund   **UID:21BCS3407**

Branch: CSE   **Section/Group: NTPP_IOT_602-B**

Semester: 5th   **Date of Performance: 22/08/23**

Subject Name: AIML   **Subject Code:21CSH-316**

1. Aim:

Implement BFS algorithm in python

2.Objective:
The objective of this experiment is to implement the Breath-First Search (DFS) algorithm and analyze its performance and characteristics

Algorithm Loop:

Step 1: SET STATUS = 1 (ready state) for each node in G

Step 2: Enqueue the starting node A and set its STATUS = 2 (waiting state)

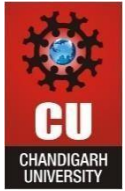Step 3: Repeat Steps 4 and 5 until QUEUE is empty

Step 4: Dequeue a node N. Process it and set its STATUS = 3 (processed state).

Step 5: Enqueue all the neighbours of N that are in the ready state (whose STATUS = 1) and set

their STATUS = 2

(waiting state)

[END OF LOOP]

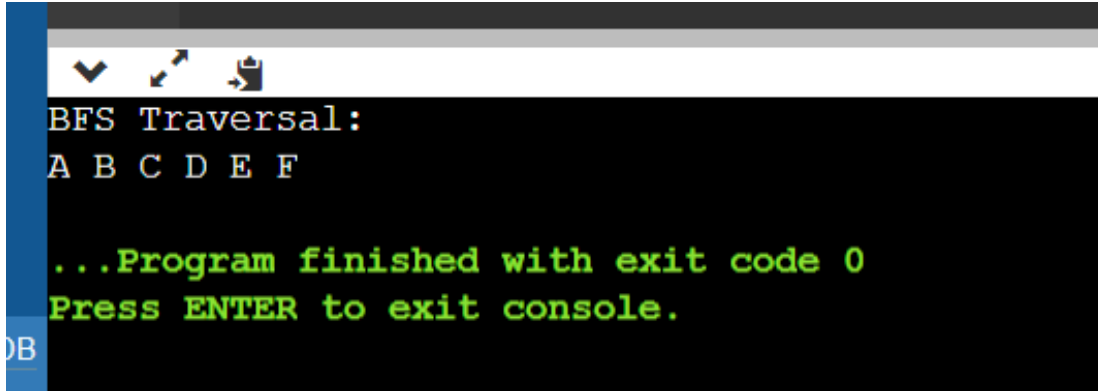Step 6: EXIT

3. Code:

```python
from collections import deque

graph = {
'A': ['B', 'C'],
'B': ['D', 'E'],
'C': ['F'],
'D': [],
'E': ['F'],
'F': []
}

def bfs(start_node):
  queue = deque([start_node])
  visited = set([start_node])
  while queue:
    node = queue.popleft()
    print(node, end=' ')
    for neighbor in graph[node]:
      if neighbor not in visited:
        queue.append(neighbor)
        visited.add(neighbor)

start_node = 'A'
print("BFS Traversal:")
```

bfs(start_node

5. Output: