Ex.No.5a
Date:4/3/2021

Best First Search Traversal

Aim: To Study and Implement Best First Search Traversal

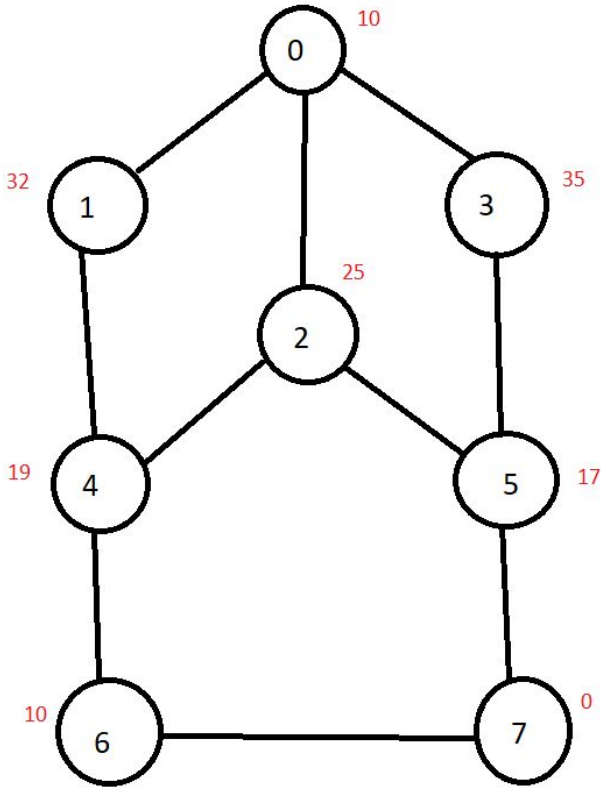
Algorithm: The whole process can be categorized as follows:
1. Create a function Called Add Edge For adding edges to the Graph.
2. Create another Function called BestFirstedge which is used calculate the Optimal Path
3. Create Two Lists one Open and One Closed.
4. Place the starting node into the OPEN list.
5. Remove the node n, from the OPEN list which has the lowest value of h(n), and places it in the CLOSED list.
6. Expand the node n, and generate the successors of node n.
7. For each successor node, algorithm checks for evaluation function f(n), and then check if the node has been in either OPEN or CLOSED list. If the node has not been in both list, then add it to the OPEN list.
8. Repeat this step till we reach The goal State
9. Print the Output.

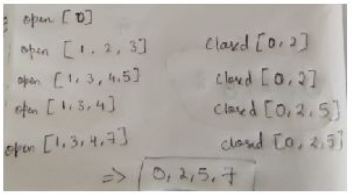
Code:

```
from queue import PriorityQueue
v = 8
graph = [[] for i in range(v)]

# Function For Implementing Best First Search
# Gives output path having lowest cost


def best_first_search(source, target, n):
    visited = [0] * n
    visited[0] = True
    pq = PriorityQueue()
    pq.put((0, source))
    while pq.empty() == False:
        u = pq.get()[1]
        # Displaying the path having lowest cost
        print(u, end=" ")
        if u == target:
            break

        for v, c in graph[u]:
            if visited[v] == False:
                visited[v] = True
                pq.put((c, v))
    print()
```

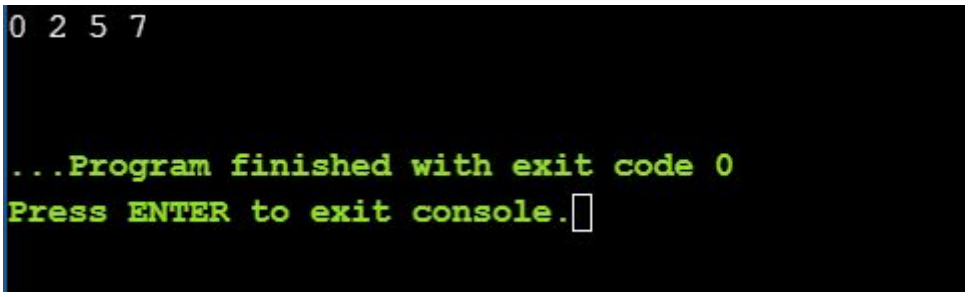
```
# Function for adding edges to graph


def addedge(x, y, cost):
    graph[x].append((y, cost))
    graph[y].append((x, cost))


# The nodes shown in above example(by alphabets) are
# implemented using integers addedge(x,y,cost);
addedge(0, 1, 32)
addedge(0, 2, 25)
addedge(0, 3, 35)
addedge(1, 4, 19)
addedge(2, 4, 19)
addedge(2, 5, 17)
addedge(3, 5, 17)
addedge(4, 6, 10)
addedge(5, 7, 0)
addedge(6, 7, 0)

source = 0
target = 7
best_first_search(source, target, v)
```

Graph:





Output:



Result: We have successfully studied and implemented Best First Search Traversal.