#Program Assignment No-1

```
graph = {  '5' : ['3','7'], '3' : ['2', '4'], '7' : ['8'], '2' : [], '4' : ['8'], '8' : []}
#Breadth-First Search
visited = [] # List for visited nodes.
queue = []     #Initialize a queue
def bfs(visited, graph, node): #function for BFS
    visited.append(node)
    queue.append(node)
    while queue:          # Creating loop to visit each node
        m = queue.pop(0)
        print (m, end = "\n")
        for neighbour in graph[m]:
            if neighbour not in visited:
                visited.append(neighbour)
                queue.append(neighbour)
#  Depth-First Search
visited1 = set() # Set to keep track of visited nodes of graph.
def dfs(visited1, graph, node):  #function for dfs
    if node not in visited1:
        print (node)
        visited1.add(node)
        for neighbour in graph[node]:
            dfs(visited1, graph, neighbour)

flag=1
while flag==1:
  print("1. Breadth-First Search \n 2. Depth-First Search\n 3. Exit\n")
  ch=int(input("Enter your Choice (from 1 to 3) :"))
  if ch==1:
    print("Following is the Breadth-First Search")
    bfs(visited, graph, '5')    # function calling
    a = input("Do you want to continue (y/n) :")
    if a == "y":
      flag = 1
    else:
      flag = 0
      print("Thanks for using this program!")
  elif ch==2:
    print("Following is the Depth-First Search")
    dfs(visited1, graph, '5')
    a = input("Do you want to continue (y/n) :")
    if a == "y":
      flag = 1
    else:
      flag = 0
      print("Thanks for using this program!")

  elif ch==3:
    flag=0
    print("Thanks for using this program!")
  else:
    print("!!Wrong Choice!! ")
    a=input("Do you want to continue (y/n) :")
    if a=="y":
      flag=1
    else:
      flag=0
```

```
print("Thanks for using this program!")
```