BFS DFS

Code

graph = {

'S': ['A', 'B'],

'A': ['C', 'D'],

'B': ['G','H'],

'C': ['E','F'], 'D': [],

'G': ['I'],

'H': [],

'E': ['K'],

'F': [],

'I': [],

'K': []

}

visitd =[]

queue=[]

visited1 =[]

def bfs(visitd,graph,node):

visitd.append(node)

queue.append(node)

while queue:

P=queue.pop(0)

print(P,end=" ")

for neighbour in graph[P]:

if neighbour not in visitd:

visitd.append(neighbour)

queue.append(neighbour)

visited = set()

def dfs(visited, graph, node):

if node not in visited:

print (node)

visited.add(node)

for neighbour in graph[node]:

dfs(visited, graph, neighbour)

print("Breadth first search")

bfs(visitd,graph,'S')

print()

print("Following is the Depth-First Search")

dfs(visited, graph, 'S')