BFS

from collections import deque

def bfs(graph, start):

visited = set()

queue = deque([start])

while queue:

vertex = queue.popleft()

if vertex not in visited:

print(vertex, end=" ")

visited.add(vertex)

queue.extend(neighbor for neighbor in graph[vertex] if neighbor not in visited)

# Get user input

graph = {}

n = int(input("Enter number of nodes: "))

for \_ in range(n):

node = input("Enter node: ")

neighbors = input(f"Enter neighbors of {node} (space separated): ").split()

graph[node] = neighbors

start\_node = input("Enter starting node for BFS: ")

print("BFS Traversal:")

bfs(graph, start\_node)

OUTPUT:

# Example Input:

# Enter number of nodes: 4

# Enter node: A

# Enter neighbors of A (space separated): B C

# Enter node: B

# Enter neighbors of B (space separated): D

# Enter node: C

# Enter neighbors of C (space separated): D

# Enter node: D

# Enter neighbors of D (space separated):

# Enter starting node for BFS: A

# Output:

# BFS Traversal:

# A B C D