

BFS 1

```
import itertools
```

```
def bfs(graph, start):
```

```
    visited = set()
```

```
    queue = [start]
```

```
    while queue:
```

```
        vertex = queue.pop(0)
```

```
        if vertex not in visited:
```

```
            visited.add(vertex)
```

```
            print(vertex) # Replace with desired operation on the vertex
```

```
            neighbors = graph[vertex]
```

```
            unvisited_neighbors = itertools.filterfalse(visited.__contains__, neighbors)
```

```
            queue.extend(unvisited_neighbors)
```

```
# User input for constructing the graph
```

```
graph = {}
```

```
n = int(input("Enter the number of vertices in the graph: "))
```

```
for i in range(n):
```

```
    vertex = input(f"Enter vertex {i + 1}: ")
```

```
    neighbors = input(f"Enter neighbors of vertex {i + 1} (space-separated): ").split()
```

```
    graph[vertex] = neighbors
```

```
start_vertex = input("Enter the starting vertex for BFS: ")
```

```
# Calling the BFS function with user-provided inputs
```

```
bfs(graph, start_vertex)
```