

✓ Nama: Dewa Bagus Putu Arya Dhananjaya

NPM: 10122362

Kelas: 3KA21

Program dalam file Jupyter Notebook (dengan nama BFS-Notebook.ipynb) merupakan implementasi dari algoritma Breadth-First Search (BFS) dalam bahasa pemrograman Python, yang digunakan untuk melakukan penelusuran atau traversal pada struktur graf. Program ini dimulai dengan mendefinisikan sebuah graf tak berarah dalam bentuk struktur data dictionary, di mana setiap simpul (node) direpresentasikan sebagai key, dan nilai dari setiap key adalah daftar simpul yang menjadi tetangga atau terhubung langsung dengannya. Misalnya, simpul 'A' memiliki tetangga 'Z', 'S', dan 'T', yang berarti dari simpul 'A', terdapat jalur langsung ke tiga simpul tersebut.

```
graph = {
    'A' : ['Z', 'S', 'T'],
    'Z' : ['O'],
    'O' : ['S1'],
    'S1' : ['F', 'R'],
    'F' : ['B'],
    'B' : [],
    'R' : [],
    'S' : ['O1', 'F1', 'R1'],
    'O1' : [],
    'F1' : ['B1'],
    'B1' : [],
    'R1' : ['C', 'P'],
    'C' : [],
    'P' : [],
    'T' : ['L'],
    'L' : ['M'],
    'M' : ['D'],
    'D' : []
}
```

Selanjutnya, program membuat dua list kosong: visited untuk mencatat node yang sudah dikunjungi, dan queue untuk mengatur antrian node yang akan dikunjungi sesuai prinsip BFS. Fungsi utama bfs() menerima parameter berupa list visited, dictionary graph, dan node awal node. Fungsi ini menambahkan node awal ke dalam list visited dan queue, kemudian menjalankan perulangan selama queue tidak kosong. Dalam setiap iterasi, node paling depan dari queue diambil (dengan operasi pop(0)), dicetak ke layar, dan seluruh tetangganya yang belum pernah dikunjungi dimasukkan ke dalam visited dan queue.

```
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 = " ")

        for neighbour in graph[m]:
            if neighbour not in visited:
                visited.append(neighbour)
                queue.append(neighbour)
```

Terakhir, fungsi BFS dipanggil dengan node awal 'A', dan hasil traversal BFS dicetak ke layar menggunakan perintah print. Traversal ini akan menjelajahi graf secara menyeluruh, mulai dari node 'A', kemudian ke semua node yang bertetangga langsung dengannya, lalu ke node-node tetangga dari hasil penelusuran sebelumnya, dan seterusnya hingga semua node terhubung dikunjungi. Program ini sangat berguna untuk memahami cara kerja algoritma BFS dalam mencari atau menjelajahi data berbasis graf secara sistematis dan mendalam.

```
# Driver Code
print("Following is the Breadth-First Search")
bfs(visited, graph, 'A')
```

```
➡ Following is the Breadth-First Search
A Z S T O O1 F1 R1 L S1 B1 C P M F R D B
```