

# Teoria dos Grafos - Busca em Profundidade e Largura

## Integrantes do Grupo:

- Igor Benites Moura - 10403462
- Rodrigo Machado de Assis Oliveira de Lima - 10401873

Implementado em Python

## Busca em profundidade:

Código fonte:

```
def dfs(self, start):  
  
    visited = [False] * self.n  
  
    stack = pilha.Pilha()  
  
    stack.push(start)  
  
    visited[start] = True  
  
    while not stack.isEmpty():  
  
        n = stack.pop()  
  
        print(f"Visitou vértice {n}")  
  
        for m in range(self.n):  
  
            if self.adj[n][m] == 1 and not visited[m]:  
  
                print(f"Vértice {n} entrou na pilha")  
  
                stack.push(n)  
  
                stack.push(m)  
  
                visited[m] = True  
  
                break
```

Testes:

```
from TGrafoComBusca import Grafo

g = Grafo(5)

g.insereA(0,1)
g.insereA(0,2)
g.insereA(1,3)
g.insereA(3,4)

# Busca em profundidade
print("Busca em profundidade")
g.dfs(0)

# Busca em largura
print("\nBusca em largura")
g.bfs(0)
```

```
Busca em profundidade
Visitou vértice 0
Vértice 0 entrou na pilha
Visitou vértice 1
Vértice 1 entrou na pilha
Visitou vértice 3
Vértice 3 entrou na pilha
Visitou vértice 4
Visitou vértice 3
Visitou vértice 1
Visitou vértice 0
Vértice 0 entrou na pilha
Visitou vértice 2
Visitou vértice 0
```

Busca em largura:

Código fonte:

```
def bfs(self, start):

    visited = [False] * self.n

    queue = fila.FilaCircular(self.n)

    queue.enqueue(start)

    visited[start] = True

    while not queue.isEmpty():

        n = queue.dequeue()

        print(f"Visitou vértice {n}")

        for m in range(self.n):
```

```
if self.adj[n][m] == 1 and not visited[m]:  
  
    print(f"Vértice {m} entrou na fila")  
  
    queue.enqueue(m)  
  
visited[m] = True
```

Testes:

```
from TGrafoComBusca import Grafo  
  
g = Grafo(5)  
  
g.insereA(0,1)  
g.insereA(0,2)  
g.insereA(1,3)  
g.insereA(3,4)  
  
# Busca em profundidade  
print("Busca em profundidade")  
g.dfs(0)  
  
# Busca em largura  
print("\nBusca em largura")  
g.bfs(0)
```

```
Busca em largura  
Visitou vértice 0  
Vértice 1 entrou na fila  
Vértice 2 entrou na fila  
Visitou vértice 1  
Vértice 3 entrou na fila  
Visitou vértice 2  
Visitou vértice 3  
Vértice 4 entrou na fila  
Visitou vértice 4
```