

Relatório - Trabalho sobre Grafos

Nome: Thiago Pereira Camerato

Matrícula: 2212580

Algoritmo MST (Árvore Geradora Mínima):

- **Funcionamento:** O algoritmo MST utiliza a estratégia Union-Find para construir uma árvore geradora mínima em um grafo não direcionado ponderado. A ideia principal é ordenar as arestas com base em seus pesos e, em seguida, adicionar as arestas à árvore, evitando a formação de ciclos.
- **Implementação no Código:**
 - A estrutura **Edge** é usada para representar arestas ponderadas.
 - A função **MST** implementa o algoritmo MST.
 - A função **find** encontra o representante (raiz) de um conjunto.
 - A função **unionSets** realiza a união de dois conjuntos.
- **Testes:**
 - O código define um grafo com 10 vértices e arestas ponderadas.
 - O algoritmo MST é aplicado ao grafo, e a árvore geradora mínima resultante é impressa.

Output :

```
Árvore Geradora Mínima (MST):  
(10, 1) Peso: 1  
(9, 8) Peso: 1  
(8, 5) Peso: 1  
(2, 10) Peso: 1  
(6, 3) Peso: 1  
(3, 4) Peso: 1  
(7, 4) Peso: 2  
(10, 8) Peso: 2  
(4, 5) Peso: 2
```

Algoritmo BFS (Busca em Largura):

- **Funcionamento:** A busca em largura é um algoritmo para percorrer ou pesquisar em estruturas de dados em largura, nesse caso, um grafo. Começando de um vértice inicial, o algoritmo visita todos os vértices vizinhos antes de se mover para os vértices mais distantes.
- **Implementação no Código:**
 - A estrutura **queueNode** é usada para representar nós de uma fila.

- A função **BFS** implementa a busca em largura.
- A função **enqueue** adiciona elementos à fila, e **dequeue** remove elementos.
- **Testes:**
 - O código solicita ao usuário um vértice inicial para a busca em largura.
 - A busca em largura é realizada a partir desse vértice, e os vértices visitados são impressos.

3. Funcionalidades e Limitações:

- **Funciona:**
 - O algoritmo MST está implementado corretamente e produz uma árvore geradora mínima para o grafo dado.
 - A busca em largura também está corretamente implementada e exibe os vértices visitados a partir de um vértice inicial válido.
- **Não Funciona:**
 - Não foram identificados problemas significativos no código que impeçam a execução correta dos algoritmos MST e BFS.
 - A entrada do usuário é tratada de maneira apropriada, garantindo que o vértice inicial da busca em largura esteja no intervalo válido.

4. Casos de Teste:

- O código foi testado com um grafo específico de 10 vértices, mas pode ser adaptado para outros grafos.
- A entrada do usuário é validada para garantir que o vértice inicial da busca em largura esteja dentro dos limites.

Conclusão: O código implementa com sucesso os algoritmos MST e BFS, proporcionando uma visão clara de como esses algoritmos funcionam. As mensagens de saída foram formatadas para facilitar a compreensão dos resultados. O código é robusto em termos de entrada do usuário e realiza as operações desejadas de forma eficiente.

Output:

```
Digite o vértice inicial para busca em largura (BFS): 2

Busca em largura (BFS) a partir do vértice 2:
Visitado 2
Visitado 10
Visitado 3
Visitado 9
Visitado 8
Visitado 1
Visitado 6
Visitado 4
Visitado 5
Visitado 7
```