**1. Algoritmo de Prim**

O algoritmo de **Prim** encontra a árvore geradora mínima em um grafo ponderado e conexo.

Texto

Descrição gerada automaticamente

**Casos de Teste**

1. Entrada Grafo:

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

1. Texto

   Descrição gerada automaticamenteSaída:

**Análise de Complexidade**

Prim

* Tempo: com matriz de adjacência (pode ser
* Espaço: para matriz de adjacência

### **Operação Básica**

No algoritmo de Prim, a operação básica ocorre quando o algoritmo escolhe o próximo vértice a ser incluído na árvore geradora mínima, baseada na menor chave (mínimo peso da aresta) de um vértice ainda não incluído na MST.

**Expressão de Custo Completa**

Somando os custos das fases descritas:

T(n) = T inicialização + T escolher mínimo + T atualização + T exibição

Substituindo os custos:

T(n) = O(n) + O(n2) + O (n2) + O (n)

Como O(n2) domina O(n), a expressão final de custo é:

T(n) = O (n2)

**Classe de Problemas**

**1. Problema de Otimização:**

* O Problema da Árvore Geradora Mínima (MST) é um problema de otimização, pois o objetivo é minimizar o custo total das arestas para conectar todos os vértices de um grafo sem formar ciclos.
* No caso do algoritmo de Prim, o problema envolve selecionar as arestas de menor peso de forma que todos os vértices sejam conectados, formando uma árvore que minimiza a soma do peso das arestas.

**2. Problema de Grafo:**

* O Problema da Árvore Geradora Mínima é resolvido em um grafo ponderado (onde as arestas têm pesos associados) e conectado (onde existe um caminho entre qualquer par de vértices).
* Esse problema pode ser formulado de forma que envolva a busca de uma subárvore com a menor soma de pesos, sem ciclos.

**3. Classe P (Problemas Resolvíveis em Tempo Polinomial):**

* O Algoritmo de Prim é um algoritmo eficiente, e o problema MST pode ser resolvido em tempo polinomial.
* O algoritmo de Prim tem complexidade O(n2) no caso de representação de grafo com matriz de adjacência (para grafos densos).

**4- Classe de Problemas Guloso:**

* O algoritmo de Prim segue uma abordagem gulosa, pois escolhe a aresta de menor peso de forma local em cada passo, buscando otimizar a solução parcial em cada iteração, sem retroceder ou revisar escolhas anteriores.
* Em cada passo, o algoritmo escolhe o vértice de menor peso que ainda não foi incluído na MST e conecta esse vértice ao conjunto de vértices já na árvore.