

Participantes: Hudson Cauã - 211055512/ Lucas Amaral 211055316

1. **Busca em largura:** Partindo de um nó inicial, são adicionados a uma fila todos os filhos do nó que está sendo visitado e para continuar a busca basta visitar os filhos com base na ordem a qual eles foram inseridos na fila, ou seja, o primeiro que entrou será o primeiro a sair.

Vantagens:

- Encontra o caminho mais curto até a solução
- Bom caso a solução não esteja longe do nó raiz

Desvantagens:

- Costuma utilizar mais memória
- Não é o ideal caso a solução esteja longe do nó raiz
- Pode demorar muito caso a árvore seja muito larga

**Busca em profundidade:** Partindo de um nó inicial, são adicionados no início de uma lista L todos os filhos desse nó, para visitar um novo nó é necessário pegar o primeiro nó da lista. Ao visitar um novo nó, todos os filhos desse novo nó serão adicionados ao começo da lista.

Vantagens:

- Utiliza menos memória
- Costuma ser mais rápida

Desvantagens:

- Pode não encontrar o caminho mais curto até a solução
- Pode demorar muito caso a árvore seja muito profunda
- Não é o ideal caso a solução não esteja longe do nó raiz

3. O problema do pico se refere aos seguintes problemas:
  - Máximo local: É possível que sejam encontrados máximos locais ao invés de globais, assim, não chegando em soluções ótimas.
  - Planície: Quando não existem muitos picos, o algoritmo tem dificuldade de fazer progresso, uma vez que o algoritmo tem como objetivo encontrar picos.
  - Picos pontiagudos: O algoritmo pode passar por picos sem perceber.
5. **Busca de custo uniforme:** se baseia em visitar o nó que possui o menor custo total para chegar ao próximo nó, ele não utiliza nenhuma heurística e sempre encontra a solução ótima para o problema. Deve ser utilizada quando é preciso encontrar a solução com o menor custo e quando não se tem conhecimento sobre o problema para formular uma heurística.

**Melhor-primeiro:** Esse algoritmo irá escolher o melhor nó para visitar baseado em uma heurística. É considerado um algoritmo guloso, pois não considera o custo total para

visitar um novo nó. Deve ser utilizado quando se quer encontrar a solução de maneira mais rápida e quando não é necessário encontrar a solução ótima.

7. Algoritmos genéticos começam com uma população inicial de indivíduos, cada um desses indivíduos são avaliados e são selecionados os mais aptos para realizar a criação de novos indivíduos por meio de operadores genéticos. Para definir quais indivíduos são aptos é utilizada uma função de aptidão e a partir dos valores retornados são escolhidos os indivíduos, essa seleção pode ocorrer por meio de torneio, roleta ou amostragem universal estocástica. Após a seleção dos indivíduos, podem ser aplicados os principais operadores genéticos: cruzamento, mutação e elitismo. Ao fim de cada nova geração são verificados os critérios de parada para decidir se uma nova geração deve ser construída.
8. Para resolver o problema do caixeiro viajante, é possível definir rotas diferentes como população inicial, a partir disso poderíamos cruzar essas soluções, dessa forma trocando a ordem das cidades visitadas e se atentando para que as novas soluções não visitem a mesma cidade mais de uma vez. Ao final de cada geração seria testado se houve melhora significativa durante um dado número de gerações.