|  |  |  |
| --- | --- | --- |
| **Universidade Federal de Sergipe – Campus São Cristóvão** | | |
| **Disciplina:** Inteligência Artificial | | **Professor:** Carlos Alberto Estombelo Montesco |
| **Atividade:** 02 – Relatório de Desempenho de Buscas não Informadas | | |
| **Alunos:** | Eduardo Fillipe da Silva Reis (201700062953);  Felipe Silva Falcão (201700053103);  Mariana Lira de Farias (201700053220); | |

1. **Breadth-first Search**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Problema** | **Tamanho Máximo da Fila** | **Nós Expandidos** | **Custo do Caminho** | **Tamanho Final da Fila** |
| **Caixeiro Viajante** | 5 | 5 | 450.0 | 5 |
| **Oito Rainhas (Incremental)** | 573 | 1665 | 8.0 \* | 301 |
| **Quebra Cabeça** | 6 | 5 | 3.0 | 6 |

*\* Não faz sentido o uso de custo nesse problema, pois sempre será o número de passos que levou até chegar no estado final, nesse caso sempre será o número* ***N*** *de rainhas.*

* 1. **Vantagens:**

Simples implementação. Sempre consegue a solução de nível mais raso em relação a profundidade da árvore, levando menos iterações para alcançá-la. É completa.

* 1. **Desvantagens:**

Não entrega necessariamente a melhor solução. Uso muito grande de memória.

1. **Uniform-cost Search**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Problema** | **Tamanho Máximo da Fila** | **Nós Expandidos** | **Custo do Caminho** | **Tamanho Final da Fila** |
| **Caixeiro Viajante** | 5 | 12 | 418.0 | 3 |
| **Oito Rainhas (Incremental)\*** | 578 | 1965 | 8.0 \* | 91 |
| **Quebra Cabeça** | 10 | 16 | 3.0 | 9 |

*\* Não há vantagem entre usar o algoritmo de BFS ou de Custo uniforme, pois o custo depende unicamente do número* ***N*** *de rainhas.*

* 1. **Vantagens:** É completa. Entrega sempre a melhor solução baseada na função de custo. Implementação relativamente simples.
  2. **Desvantagens:** Uso muito grande de memória.

1. **Depth-first search**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Problema** | **Tamanho Máximo da Fila** | **Nós Expandidos** | **Custo do Caminho** | **Tamanho Final da Fila** |
| **Caixeiro Viajante** | 6 | 11 | 575.0 | 4 |
| **Oito Rainhas (Incremental)\*** | 19 | 113 | 8.0 \* | 11 |
| **Quebra Cabeça** | 337 | 438 | 431.0 | 336 |

*\* Não faz sentido o uso de custo nesse problema, pois sempre será o número de passos que levou até chegar no estado final, nesse caso sempre será o número* ***N*** *de rainhas.*

* 1. **Vantagens:** Evita estados repetidos e caminhos redundantes. É completa, por haver estados finitos (expande cada nó).
  2. **Desvantagens:** Explora caminhos desnecessários.

**4. Depth-limited search**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Problema** | **Tamanho Máximo da Fila** | **Nós Expandidos** | **Custo do Caminho** | **Tamanho Final da Fila** |
| **Caixeiro Viajante** | - | 10 | 1290.0 | - |
| **Oito Rainhas (Incremental)\*** | - | 113 | 8.0 | - |
| **Quebra Cabeça** | - | 104 | 99.0 | - |

* 1. **Vantagens:** Se comparada a Depth-first search resolve o problema de espaços infinitos, porém também não é completa e nem ótima.
  2. **Desvantagens:** É necessário conhecer um bom limite de profundidade para resolver o problema e nem sempre isso é possível, pois cada problema apresenta um valor particular e não existe um padrão para encontrá-lo.

**5. Iterative Deeping Depht-First Search**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Problema** | **Tamanho Máximo da Fila** | **Nós Expandidos** | **Custo do Caminho** | **Tamanho Final da Fila** |
| **Caixeiro Viajante** | - | 9.0 | 450.0 | - |
| **Oito Rainhas (Incremental)\*** | - | 3656.0 | 8.0\*\* | - |
| **Quebra Cabeça** | - | 9.0 | 3.0 | - |

*\* Não faz sentido o uso de custo nesse problema, pois sempre será o número de passos que levou até chegar no estado final, nesse caso sempre será o número* ***N*** *de rainhas.*

* 1. **Vantagens:** É completa, já que os fatores de ramificação são finitos. Sua implementação é relativamente simples já que somente aplica repetidas vezes (com limites de profundidade crescentes) o algoritmo de busca por profundidade.
  2. **Desvantagens:** Repetição de busca, gerando maior consumo de memória.