

Inteligência Computacional

Rafael D. Ribeiro, M.Sc.
rafaeldiasribeiro@gmail.com
<http://www.rafaeldiasribeiro.com.br>

Inteligência Computacional

Métodos Revogáveis de Busca

- Busca em profundidade
- Backtracking
- **Busca em largura**
- Busca em profundidade iterativa
- Busca ordenada

Inteligência Computacional

Busca em largura

- A busca em largura prioriza os nós mais próximos do nó raiz, isto é, os de menor profundidade. Ela expande os nós na ordem em que são gerados fazendo com que os nós de uma determinada profundidade somente são gerados e avaliados se os da profundidade anterior já tiverem sido abordados.

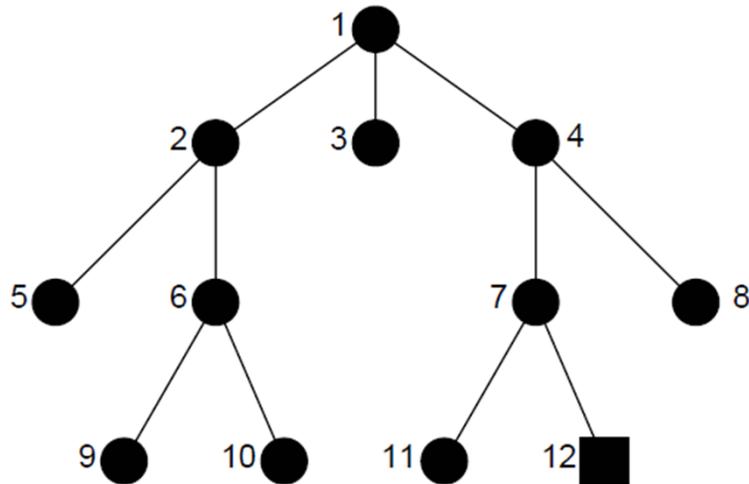
Inteligência Computacional

Busca em largura

- Inicialmente, o nó raiz é expandido e seus filhos avaliados. Se nenhum corresponder a alguma solução, então **o filho mais à esquerda é expandido**.
- Se os filhos deste também não representarem alguma solução, então **o segundo filho do nó raiz é expandido**. E assim o método continua expandindo os filhos do nó raiz até encontrar alguma solução ou não haver mais filhos a serem expandidos.
- Se nenhuma solução for encontrada na profundidade 2, então os nós da profundidade 3 passam a ser expandidos **repetindo-se todo o procedimento de expansão e avaliação nível a nível**.

Inteligência Computacional

Busca em largura



Inteligência Computacional

Busca em largura

- A busca em largura avalia todos os ramos de uma árvore. Assim, além de garantir o encontro de uma solução se ela existir, esta solução é ótima. Portanto, **é uma busca completa e ótima.**
- **Desvantagem:** Há uma grande necessidade de memória para armazenar toda a árvore.

Inteligência Computacional

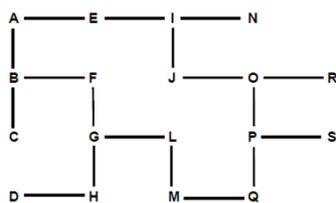
Algoritmo Básico da Busca em Largura

```

resposta = nulo
fila-de-abertos = estado inicial
lista-de-fechados = nulo
sucesso = falso
enquanto (sucesso = falso) e (fila-de-abertos ≠ vazio) faça
    nó-candidato = elemento do início da fila-de-abertos
    remova o elemento do início da fila-de-abertos
    coloque-o em lista-de-fechados
    expanda nó-candidato
    se algum filho de nó-candidato é a solução
        então
            sucesso = verdadeiro
            resposta = filho-solução de nó-candidato
        senão
            coloque o(s) nó(s) filho(s) ao final da fila-de-abertos
    fim-se
fim-enquanto
retorna sucesso e resposta
  
```

Inteligência Computacional

Problema do Labirinto



Método Busca em Largura

Entrada: A Saída: S

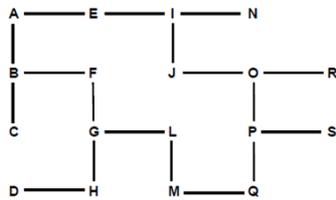
Regras:

Regra 1: ↓ Regra 2: ←
 Regra 3: ↑ Regra 4: →

Estratégia:
 As regras são aplicadas na ordem acima

Inteligência Computacional

Problema do Labirinto



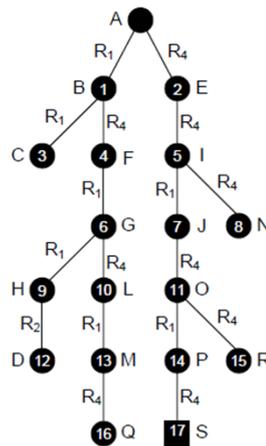
Entrada: A Saída: S

Regras:

Regra 1: ↓ Regra 2: ←
 Regra 3: ↑ Regra 4: →

Estratégia:
 As regras são aplicadas na ordem acima

Método Busca em Largura



Inteligência Computacional

Problema dos jarros de água

- Existem dois jarros inicialmente vazios. Um possui capacidade igual a 3 litros e o outro igual a 4 litros.
- Ambos podem ser enchidos completamente utilizando uma torneira e podem também ser esvaziados, despejando a água em um ralo.
- Além disso, a água presente em um jarro pode ser passada para o outro.
- Os jarros não possuem marcações e não é permitido o uso de qualquer instrumento de medida. Deseja-se colocar exatamente dois litros de água no jarro maior.
- Pode-se representar as quantidades de água presentes nos dois jarros pelo par ordenado (x,y) , em que x é a quantidade de água no jarro menor e y a quantidade no jarro maior.
- O estado inicial (ambos vazios) é o par $(0,0)$ e o objetivo é encontrar um par do tipo $(x,2)$, isto é, dois litros no jarro maior e qualquer quantidade no jarro menor.

Inteligência Computacional

Problema dos jarros de água

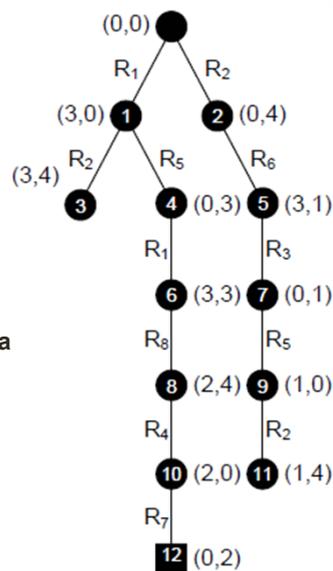
- As regras que definem as ações permitidas são exibidas a seguir, observando que a estratégia de aplicação das mesmas é a ordem de apresentação.

R1: Se $x < 3$, então $(3,y)$. // regra para encher o jarro menor
 R2: Se $y < 4$, então $(x,4)$. // regra para encher o jarro maior
 R3: Se $x > 0$, então $(0,y)$. // regra para esvaziar o jarro menor
 R4: Se $y > 0$, então $(x,0)$. // regra para esvaziar o jarro maior
 R5: Se $y > 0$ e $x+y \leq 3$, então $(x+y,0)$. // regra para passar toda a água do jarro maior para o menor
 R6: Se $y > 0$ e $x+y > 3$, então $(3,y - (3-x))$. // regra para passar parte do maior para o menor
 R7: Se $x > 0$ e $x+y \leq 4$, então $(0,x+y)$. // regra para passar toda a água do jarro menor para o maior
 R8: Se $x > 0$ e $x+y > 4$, então $(x - (4-y),4)$. // regra para passar parte do menor para o maior

Inteligência Computacional

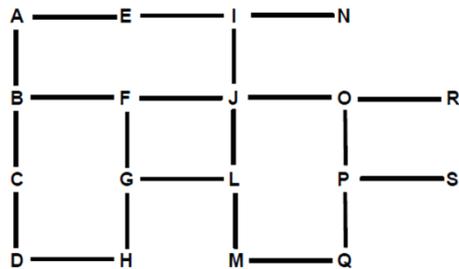
Problema dos jarros de água

Solução – Método Busca em Largura



Inteligência Computacional

Utilizando o Método de Busca em Largura , resolva:



Regras

Regra 1: →

Regra 2: ↓

Regra 3: ↑

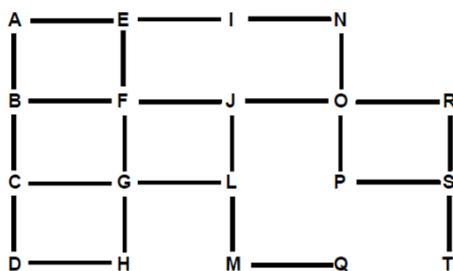
Regra 4: ←

Partida: N
Destino: Q

Estratégia:
As regras são aplicadas na ordem acima

Inteligência Computacional

Utilizando o Método de Busca em Largura , resolva:



Regras

Regra 1: →

Regra 2: ↓

Regra 3: ↑

Regra 4: ←

Partida: A
Destino: Q

Estratégia:
As regras são aplicadas na ordem acima

Inteligência Computacional

- Bibliografia utilizada para estas notas de aula:
 - Russel, Stuart J. Inteligência Artificial. Stuart J. Russel. Rio de Janeiro. Elsevier. 2004
 - Notas de Aula do Prof. ROGÉRIO ESPÍNDOLA, disponível na Biblioteca Virtual de docentes – SIA – Estácio, para a disciplina de Inteligência Computacional 1