

Inteligência Artificial

Primeiro Trabalho - Algoritmos de Busca

Profa. Heloisa

2018/2

Este trabalho consta da execução dos algoritmos de busca informada (busca gulosa, algoritmo A) estudados na disciplina, com o uso das implementações em R disponibilizadas no repositório [basicAI_Search](#) e descritas neste documento.

O repositório contém: implementações em R de todos os algoritmos, que retornam o caminho (solução) encontrado durante o processo de busca; o script Estado.R com definições gerais sobre a representação do problema; um projeto R chamado classAI.RProj; um script com a representação específica para o problema dos Missionários e Canibais para auxiliar a compreensão e definição exigida para a correta execução dos algoritmos; um script inicial chamado exemploCanibais.R onde são definidos os estados iniciais e finais e carregados os algoritmos para execução.

Para este trabalho, podem ser utilizadas as implementações dos algoritmo de busca, o script Estado.R e o projeto classAI.RProj. As representações específicas dos problemas solicitadas deverão ser definidas e utilizadas para as execuções dos algoritmos. Nessas representações devem ser incluídas: representação dos estados, operadores de mudança de estado, função heurística, função de custo. O script inicial também deve ser definido de acordo com os problemas e algoritmos solicitados.

Os algoritmos devem ser aplicados a dois problemas, definido a seguir:

- 1) **Problema do quebra-cabeça de oito peças**, com função heurística $h = \text{soma das distâncias de cada peça fora do lugar para sua posição correta}$ (distância de Manhattan). Considerar que todos os movimentos têm custo unitário.
- 2) **Problema do trajeto entre duas cidades**, considerando o mapa de cidades da Romênia dado em aula, com função heurística $h = \text{distância em linha reta entre cada cidade e a cidade destino}$. Considerar como função de custo o comprimento real, em quilômetros, da estrada que liga duas cidades. O mapa pode ser reduzido com a eliminação das cidades N, I, V, H, E. Definir um problema (cidade inicial e cidade final) de tal forma que as soluções encontradas pelos algoritmos de Busca Gulosa e A* sejam diferentes.

Relatório

O trabalho deve ser acompanhado de um relatório que documente claramente como os problemas foram formulados e como os itens dessas formulações foram definidos e representados na implementação. Os principais itens que devem ser descritos são: a representação do estado, os operadores aplicáveis aos estados (ações), os custos dos operadores, a função heurística, o estado inicial e o estado objetivo. A qualidade e o conteúdo do relatório são levados em consideração na avaliação do trabalho.

Observações

- O trabalho pode ser feito em duplas;
- Entregar (no ava – tarefa de arquivo único):
 - Relatório, com o conteúdo detalhado anteriormente;
 - Scripts em R específicos definidos para os dois problemas mencionados;
 - Script em R para inicialização (definição dos estados iniciais e finais e execução dos algoritmos);

- Saídas dos dois algoritmos para situações específicas dos dois problemas, de acordo com o que foi solicitado no enunciado.
 - **DATA DE ENTREGA:** 08/10/2018.
-

Implementações Disponíveis

O repositório [basicAI_Search](#) contém os scripts em R que serão necessários para a execução do trabalho de Inteligência Artificial sobre busca informada.

São necessários os arquivos `Estado.R` e `buscaInformada.R` (algoritmos greedy e A*).

Seu trabalho corresponde à implementação dos métodos genéricos definidos em `Estado.R` e Sobrecarga do operador `==`, que define se dois estados são iguais, para os problemas específicos **Problema do quebra-cabeça de oito peças** e **Problema do trajeto entre duas cidades**.

Missionários e Canibais

O arquivo `Canibais.R` traz uma implementação para o problema de 3 Missionários e 3 Canibais, definindo a geração de novos estados considerando um conjunto de operadores e a avaliação do valor de heurística para esse problema.

Um script com a instanciação do problema dos Missionários e Canibais e execução dos algoritmos de busca também está disponível no repositório (arquivo `exemploCanibais.R`).

Inicialização

Definição dos nós inicial e objetivo:

```
inicial <- Canibais(desc = c(M = 3, C = 3, B = 1))

objetivo <- Canibais()
objetivo$desc <- c(M = 0, C = 0, B = 0)
```

Busca Informada

Para a busca informada, o algoritmo utilizado é o *best first*. A execução do *best first* exige a passagem de um valor para o parâmetro `abordagem`, além dos nós inicial e objetivo. O valor do parâmetro deve ser uma string indicando qual cálculo da função de avaliação deve ser utilizado: como definido para o algoritmo “Greedy” ou como definido para o algoritmo “AEstrela”.

A string “AEstrela” foi definida como padrão, então se não for passado valor para o parâmetro `abordagem`, o *best first* utilizará a função de avaliação definida para o A*.

```
buscaBestFirst(inicial, objetivo, "AEstrela")
```

```
## [[1]]
## (M C B): ( 3 3 1 )
## G(n): 0
## H(n): Inf
## F(n): Inf
##
## [[2]]
## (M C B): ( 3 1 0 )
## G(n): 1
```

```

## H(n): 4
## F(n): 5
##
## [[3]]
## (M C B): ( 3 2 1 )
## G(n): 2
## H(n): 6
## F(n): 8
##
## [[4]]
## (M C B): ( 2 1 0 )
## G(n): 3
## H(n): 3
## F(n): 6
##
## [[5]]
## (M C B): ( 2 2 1 )
## G(n): 4
## H(n): 5
## F(n): 9
##
## [[6]]
## (M C B): ( 2 0 0 )
## G(n): 5
## H(n): 2
## F(n): 7
##
## [[7]]
## (M C B): ( 3 0 1 )
## G(n): 6
## H(n): 4
## F(n): 10
##
## [[8]]
## (M C B): ( 1 0 0 )
## G(n): 7
## H(n): 1
## F(n): 8
##
## [[9]]
## (M C B): ( 2 0 1 )
## G(n): 8
## H(n): 3
## F(n): 11
##
## [[10]]
## (M C B): ( 0 0 0 )
## G(n): 9
## H(n): 0
## F(n): 9

```

Nota: Vale lembrar que o problema com 3 Missionários e 3 Canibais possui uma árvore de busca pequena e simples, então o caminho retornado para todos os algoritmos de busca é o mesmo.