

**UNIVERSIDADE ESTADUAL DE PONTA GROSSA
SETOR DE CIÊNCIAS AGRÁRIAS E DA TECNOLOGIA
DEPARTAMENTO DE INFORMÁTICA**

**MATEUS FELIPE DA SILVA JUNGES
VICTOR PIOTROVSKI BEGHA**

**INTELIGÊNCIA COMPUTACIONAL:
BUSCA NÃO INFORMADA**

**PONTA GROSSA
2018**

1 QUESTÕES

1) Defina formulação do problema e formulação de objetivos (goal formulation, problem formulation).

A formulação de objetivos se refere à delimitação de metas que o agente deve tentar atingir, de forma que o comportamento do agente deve ser organizado com base no ambiente atual da situação dada, e do desempenho que o agente pode ter dentro dela; por exemplo, se existe o objetivo de chegar em determinada cidade, cursos de ação que não chegam nesta cidade podem ser desconsiderados, ou seja, a escolha de ações é simplificada a partir do delineamento daquilo que o agente está buscando alcançar.

Formulação de um problema é, essencialmente, o procedimento de decidir quais serão os estados e ações que serão considerados dentro do contexto do agente, isto é, dentre todas as ações e estados que poderiam existir, quais são relevantes para o tratamento do problema em questão. Como exemplo, pode-se considerar o problema de movimentação entre cidades, onde um estado poderia ser considerado a cidade atual, e as ações seriam as possíveis movimentações entre as cidades; tal que detalhes irrelevantes como a movimentação do veículo dentro da cidade não são fatores que fazem parte da busca do objetivo principal e assim são deixados de fora do problema.

2) Quais os elementos de uma definição de problemas e soluções? Qual o papel da abstração na formulação de um problema?

A definição de problemas e soluções é composta por vários fatores: um estado inicial (que é a situação onde o agente inicia seu comportamento); um conjunto de ações possíveis (que o agente pode executar em determinado momento); um modelo de transição para cada ação (ou seja, uma descrição do que cada ação faz e quais serão suas consequências); um teste de objetivo (verificando se o estado atual cumpre o objetivo do problema); e uma função do custo de

caminho, que confere um valor numérico a cada caminho capaz de alcançar a solução.

A abstração essencialmente envolve remover detalhes da representação do problema, tal que apenas as informações relevantes para atingir o objetivo façam parte da situação. Tanto a descrição do estado como as ações individuais são abstraídas: por exemplo, no caso do problema de rotas entre cidades, o estado é abstraído ao desconsiderar fatores da viagem como condições da estrada ou do veículo, considerando apenas as distâncias entre cada cidade; da mesma forma, as ações são abstraídas ao ignorar os detalhes menores e considerar apenas o movimento de uma cidade até outra. A abstração tem por objetivo tornar o problema o mais simples possível e ainda manter a validade das ações tomadas e da solução.

3) Qual a diferença entre usar uma estrutura de dados do tipo árvore para representar buscas e usar um grafo?

Enquanto todas os algoritmos de busca seguem o mesmo conceito básico - seguir pelos nós e verificar se a opção atual é a solução, deixando para verificar as outras caso não seja a solução - ao usar uma estrutura de dados do tipo árvore, existe a possibilidade de que surjam caminhos em laço, ou seja, caminhos redundantes onde existem várias formas de chegar em um mesmo estado; em comparação, na busca em grafo, o algoritmo guarda a informação dos nós que já foram expandidos através de uma estrutura que armazena o conjunto explorado, possibilitando descartar nós que já foram gerados se já se conhece um caminho mais eficiente para alcançá-los.

4) Defina os seguintes conceitos relacionados a algoritmos de busca:

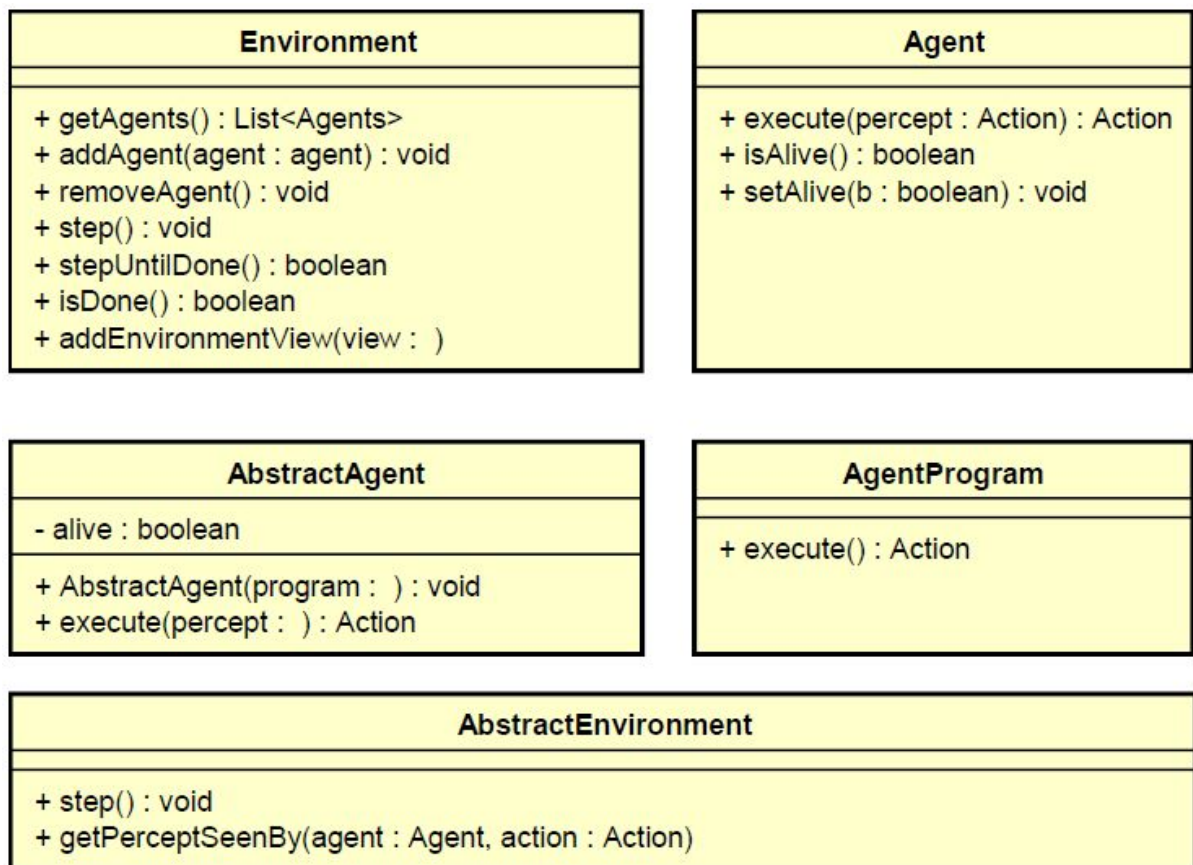
- **completeza:** A completeza se refere à capacidade do algoritmo de garantir que pode encontrar uma solução para o problema, no caso de que exista solução. Ou seja, a estratégia aplicada pelo algoritmo deve

conseguir obter a solução através da busca para todo problema que seja solucionável.

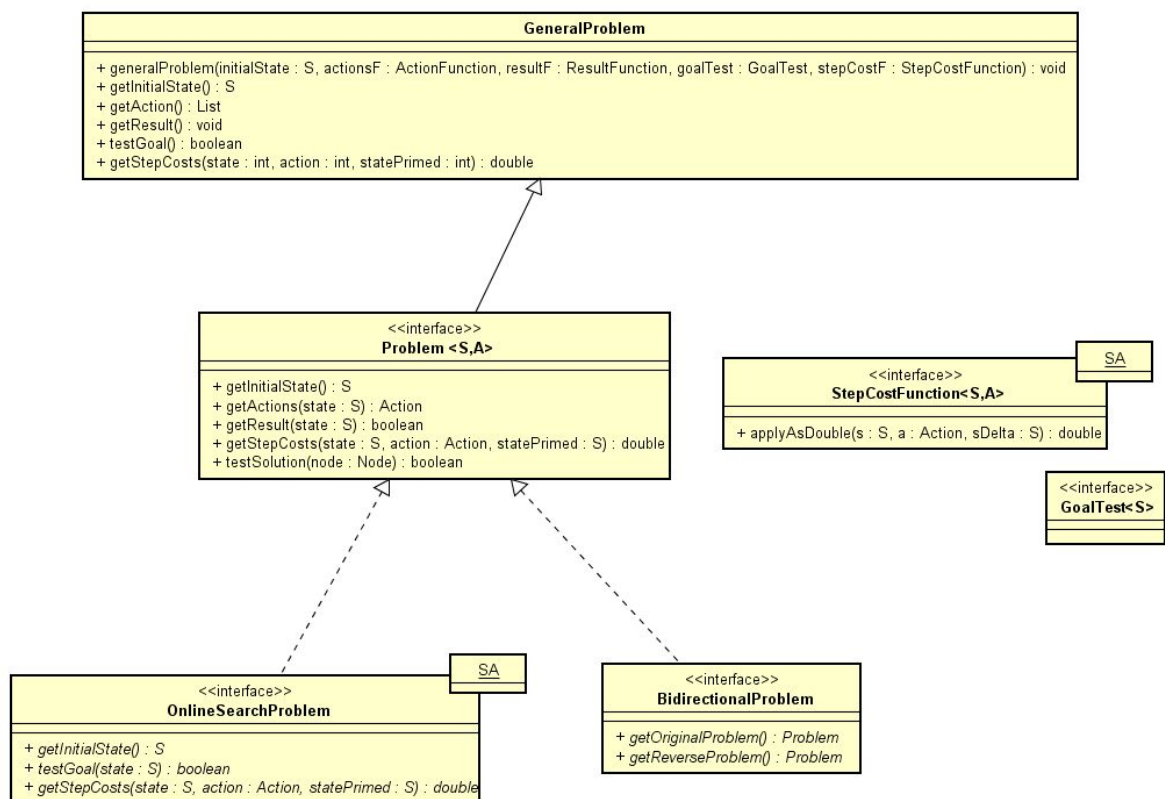
- **otimalidade:** Otimalidade mede a capacidade do algoritmo de busca em encontrar a solução ótima em uma situação em que existem várias soluções possíveis; isso avalia o desempenho do algoritmo em encontrar a solução mais eficiente para o problema.
- **complexidade de tempo:** Complexidade de tempo verifica simplesmente quanto tempo o algoritmo precisa para encontrar uma solução, ou seja, a duração da busca.

2 BIBLIOTECA AIMA

1) Elabore um diagrama de classes do pacote `aima.core.agent`.



2) Elabore um diagrama de classes e descreva o funcionamento da função *search* nas classes *aima.search.framework.Problem*, *aima.core.search.uninformed.DepthLimitedSearch* e *aima.core.search.uninformed.DepthFirstSearch*. Se for necessário inclua as classes diretamente associadas a elas (super classes e importações) no diagrama e na descrição.



DepthFirstSearch: A pesquisa em profundidade expande o nó mais profundo do na fronteira atual da árvore de pesquisa.

DepthLimitedSearch: Retorna uma lista de ações para alcançar o objetivo, além de dizer se foi encontrado ou está vazio. A lista pode estar vazia se o estado inicial for o estado de meta

