

Inteligência Artificial

Profa. Dra. Viviane Menezes

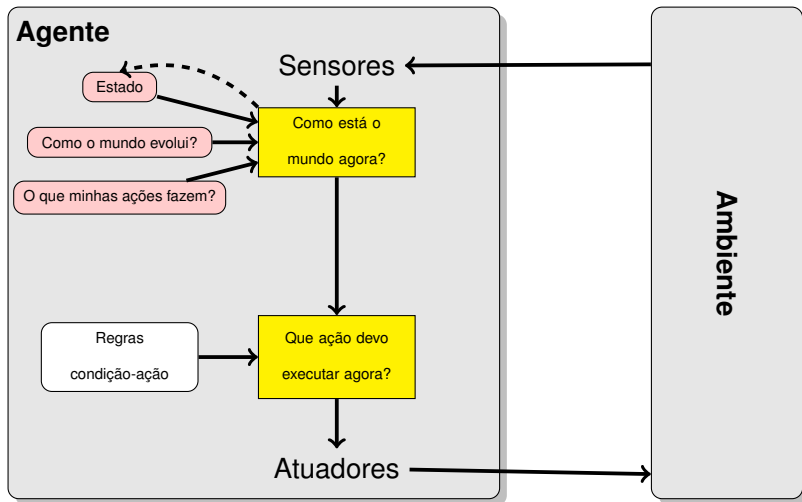
vivianemenezes@ufc.br



UNIVERSIDADE
FEDERAL DO CEARÁ
Campus Quixadá

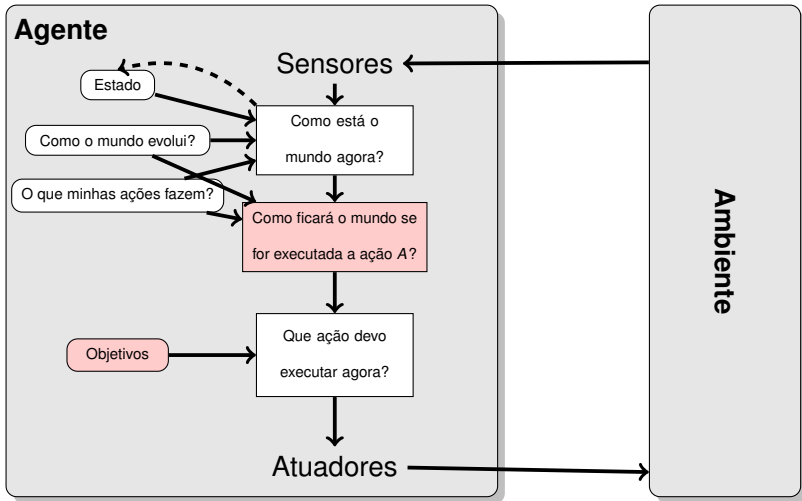
Agentes de Resolução de Problemas

Agentes de Resolução de Problemas



Agentes Baseados em Modelos.

Agentes de Resolução de Problemas



Agentes Baseados em Objetivo.

Agentes de Resolução de Problemas

- **Agentes Baseados em Objetivos.**
 - Utiliza a **representação atômica** dos estados do mundo.
-
- *Como definir um problema?*
 - *Como definir um objetivo?*
 - *Como consstruir algoritmos que alcance os objetivos definidos?*

Agentes de Resolução de Problemas

■ **Formulação de problemas**

■ **Busca por uma solução**

Formulação de Problemas

Um **problema** é definido pelos seguintes componentes:

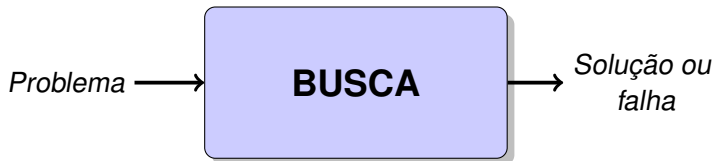
- 1 o **estado inicial** em que o agente se encontra.
- 2 Uma descrição das **ações** que o agente é capaz de executar.
 - $ACOES(s)$: conjunto de ações que podem ser executadas em s .
- 3 Um **modelo de transição de estados**
 - $RESULTADO(s, a)$
- 4 O **teste de objetivo**
 - determina se um estado é um estado objetivo.
- 5 **Custo de caminho**
 - soma dos custos das ações individuais ao longo do caminho.

Atividade Assíncrona - Formulação de Problema

- Implemente a formulação do problema do Mapa da Romênia conforme instruções do roteiro.
- Reunião com as duplas!

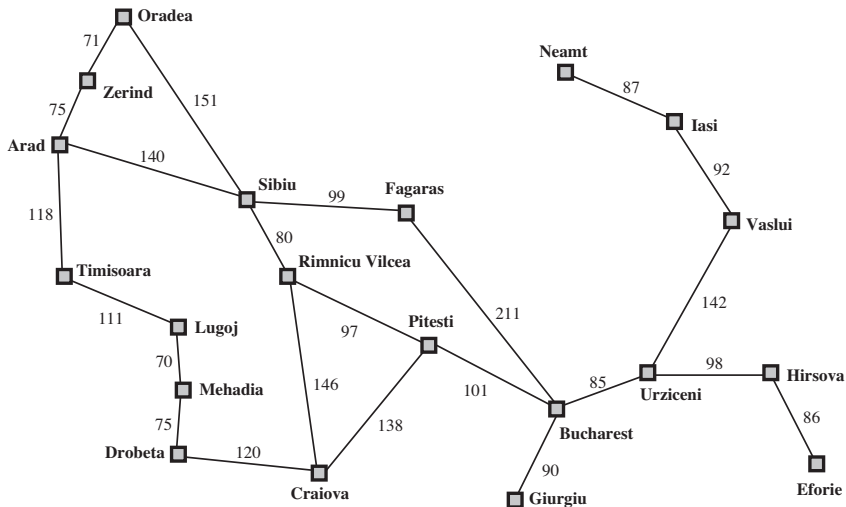
Em Busca de uma Solução

Em Busca de uma Solução



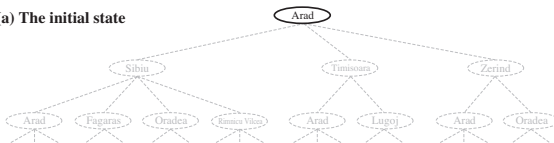
Construção de uma **árvore de busca**: os nós são formados pelos estados do problema; os ramos são as ações.

Em Busca de uma Solução

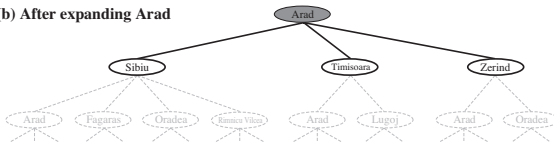


Em Busca de uma Solução

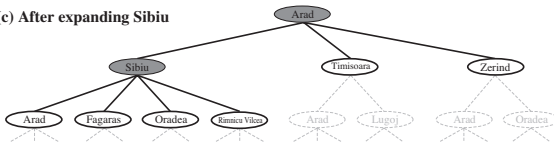
(a) The initial state



(b) After expanding Arad



(c) After expanding Sibiu



Em Busca de uma Solução

```
01. BUSCA(problema){  
02.   iniciar borda com estado inicial  
03.   repita  
04.     se borda está vazia  
05.       retorne falha  
06.     nó ← remover um nó da borda  
07.     se nó contém um estado objetivo então  
08.       retorne solução  
09.     expandir o nó escolhido.  
10. }
```

■ ***borda***: nós aguardando serem visitados.

Em Busca de uma Solução

```
01. BUSCA(problema){  
02.   iniciar borda com estado inicial  
03.   iniciar explorados com vazio  
03.   repita  
04.     se borda está vazia  
05.       retorne falha  
06.     nó ← remover um nó da borda  
07.     se nó contém um estado objetivo então  
08.       retorne solução  
09.     adicionar nó a explorados.  
09.     expandir o nó se não estiver na borda ou em explorados.  
10. }
```

- ***borda***: nós aguardando serem visitados.
- ***explorados***: **estados** já visitados.

Em Busca de uma Solução

Árvore de Busca: Estrutura Nó

- **Estado:** o estado a que o nó corresponde;
- **Pai:** o nó na árvore que gerou esse nó;
- **Ação:** a ação que foi aplicada ao pai para gerar o nó;
- **Custo do Caminho:** o custo de sair do estado inicial e alcançar o nó.

Em Busca de uma Solução

Algoritmos de Busca

- Busca em Largura
- Busca de Custo Uniforme
- Busca em Profundidade

As estratégias diferenciam-se na forma como a fila *da borda* é implementada.

Em Busca de uma Solução

Algoritmos de Busca

- Busca em Largura
- Busca de Custo Uniforme
- Busca em Profundidade

Busca em Largura

- Expande os nós, nível a nível.
- A **borda** é uma fila FIFO (*First In First Out*)

Busca em Largura

```
01. BUSCA-EM-LARGURA(problema){
02.   /* borda com nó.estado inicial e explorados vazio */
03.   repita
04.     se borda está vazia
05.       retorne falha
06.     nó ← remover elemento da borda
07.     adicionar nó.estado a explorados
08.     para cada ação aplicável em nó.estado
09.       filho ← criar nó filho
10.       se filho.estado não está em explorados ou borda
11.         se filho.estado é objetivo
12.           retorne solução
13.         adicionar filho em borda
13. }
```

Atividade Assíncrona

- Implementação da busca em Largura seguindo o roteiro disponível no Moodle.