

# QXD0037 - Inteligência Artificial

## Laboratório 01 - Agentes Inteligentes

Profa. Dra. Viviane Menezes

21 de agosto de 2018

### 1 Objetivos

Os objetivos desta atividade prática são: conhecer programas de agentes e implementar os programas de *agentes dirigido por tabela* e *agente reativo simples* para o mundo do aspirador de pó com duas salas.

### 2 Regras

- A atividade deve ser feita em dupla.
- Cada dupla deve entregar um único arquivo compactado (formato zip).
- O programa deve ser escrito em uma linguagem orientada a objetos.

### 3 Agentes e Ambientes

Um agente percebe seu ambiente por meio de *sensores* e efetua ações sobre este ambiente por meio de seus *atuadores*. No mundo do aspirador de pó, ilustrado na Figura 1, temos duas salas: sala A e sala B. O agente aspirador de pó é capaz de perceber em que quadrado está e se existe sujeira no quadrado. O agente pode optar por mover-se para *direita*, mover-se para *esquerda* ou *aspirar* a sujeira [Russell and Norvig, 2010].

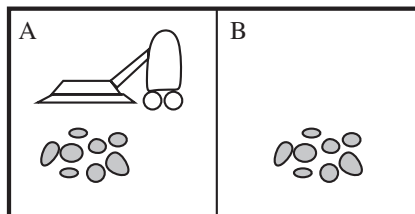


Figura 1: O mundo do aspirador de pó com duas salas.

## 4 Programas de Agentes

Um programa de agente recebe como entrada a percepção atual proveniente dos sensores e devolve uma ação para os atuadores. Cada tipo de programa de agente combina componentes específicos de maneiras específicas para gerar ações.

A Figura 2 ilustra o pseudocódigo do *programa de agente dirigido por tabela*. A cada nova percepção do agente, o programa deve retornar uma dada ação, conforme indicado na tabela de percepções do agente. Este programa mantém a sequência de percepções na memória.

```
01. AGENTE-DIRIGIDO-POR-TABELA(percepção){  
02.     variáveis:  
03.         percepções, uma sequência inicialmente vazia  
04.         tabela, tabela completa percepções-ação  
05.     anexar percepção ao fim de percepções  
06.     ação ← acessar(percepções, tabela)  
07.     retorne ação  
08. }
```

Figura 2: Pseudocódigo de um programa de agente dirigido por tabela.

A Figura 3 ilustra o pseudocódigo do *programa de agente reativo simples*. Segundo este programa, o agente age de acordo com uma regra cuja condição corresponde ao estado atual definido pela percepção.

```
01. AGENTE-REATIVO-SIMPLES(percepção){  
02.     variáveis:  
03.         regras, um conjunto de regras condição-ação  
04.     estado ← interpretar-entrada(percepção)  
05.     regra ← regra-correspondente(estados, regras)  
06.     ação ← ação-da-regra(regra)  
07.     retorne ação  
08. }
```

Figura 3: Pseudocódigo de um programa de agente reativo simples.

## 5 Implementação

Você deve implementar os programas de agente baseado em tabela e agente reativo simples para o mundo do aspirador de pó, seguindo as seguintes especificações:

- Nome do Projeto: VacuumWorld

- Classes:

- Perception:

- \* `location`, variável booleana que é verdadeira quando o robô está na sala A e falsa quando o robô está na sala B.
    - \* `isDirty`, variável booleana que é verdadeira se a sala na qual o robô se encontra está suja e falsa caso a sala que o robô se encontra esteja limpa.

- Action

- \* `name`, string que armazena o nome de uma ação do agente.

- Agent

- \* `perception`, variável que irá armazenar a percepção atual do agente proveniente do sensor de localização e verificação de sujeira.
    - \* `actions`, conjunto de ações que o agente pode executar.

A classe `Agent` terá duas especializações `TableDrivenAgent` (agente dirigido por tabela) e `SimpleReflexAgent` (agente reativo simples) nas quais serão implementadas as especificidades de cada programa de agente.

- Environment

- \* `isDirtyA` variável booleana que é verdadeira se a sala A está limpa.
    - \* `isDirtyB` variável booleana que é verdadeira se a sala B está limpa.
    - \* `agentLocation` variável booleana que é verdadeira se o agente está na sala A e falsa se o agente está na sala B.

- Main - classe com a função principal do programa.

Seu programa deve receber uma configuração inicial do ambiente e uma quantidade  $n$  de passos e deve mostrar na tela em cada um dos  $n$  passos:

- A ação executada pelo agente
- A situação do ambiente após executada tal ação.

## Referências

[Russell and Norvig, 2010] Russell, S. and Norvig, P. (2010). *Artificial Intelligence*. Elsevier, 3a edition.