



Disciplina: Inteligência Artificial

Professor: Vasco Furtado

Datas de entrega do relatório e apresentação: 2 de Abril

Orientações: Equipes de, no máximo, 2 pessoas. O não-comparecimento de um dos integrantes do grupo na respectiva data (e horário) de apresentação implicará em nota “zero”. Eventualmente, alguns *bugs* do simulador podem ser encontrados; nesse caso, mensagens serão enviadas em tempo e novas versões serão colocadas no repositório da disciplina.

ATIVIDADE COMPUTACIONAL 1

A proposta desta atividade computacional é a de avaliar a habilidade dos alunos da Disciplina de Inteligência Artificial da UNIFOR com respeito à especificação e implementação de programas de agentes inteligentes (racionais). No caso, os agentes a serem implementados devem ser agentes de *software* simples representando personagens em um jogo de computador denominado poupador. Para tanto, será disponibilizado aos alunos um ambiente de simulação completo na forma de um *framework* computacional em Java 1.5 (ver Fig. 5), em que a maior parte do jogo (i.e., parte gráfica do ambiente físico, sensores e atuadores dos agentes, etc.) já estará implementada. Caberá aos grupos de alunos somente projetar e implementar a parte comportamental do agente (poupador ou ladrão), ou seja, implementar o que seria a ação do agente, de acordo com um dos tipos básicos: reativo simples, reativo com modelo do mundo, baseado na utilidade. A avaliação da atividade computacional será feita tanto pela análise do relatório gerado pelo grupo como por arguição oral envolvendo todos os membros do grupo em uma data a ser programada. O relatório deverá ser sintético, com no máximo três páginas, contendo as principais decisões de projeto e implementação e justificando a escolha do modelo de programa de agente adotado.

Seguem abaixo outras informações importantes acerca da atividade computacional:

- O ambiente físico (Fig. 1) consistirá de um labirinto, composto por uma série de obstáculos (paredes), por moedas espalhadas aleatoriamente, pastilhas de poder, agências bancárias, um agente poupador e um agente ladrão. O labirinto deve ser visto como uma matriz de células de tamanho idêntico. A configuração física do labirinto será diferente no dia da avaliação.

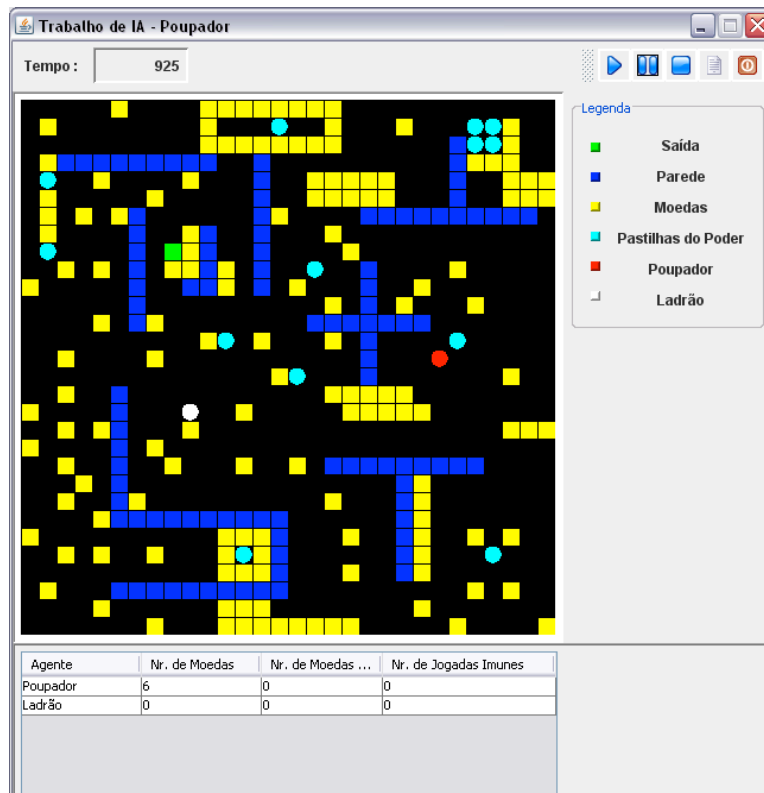


Figura 1- Exemplo de configuração do ambiente do jogo

- Dois tipos de agentes habitam o labirinto: “agente-poupador” e “agente-ladrão”. Deverão existir mais de um agente agente-poupador e mais de um agente ladrão por partida. Alguns grupos de alunos ficarão responsáveis pela implementação do agente-poupador, enquanto outros grupos ficarão responsáveis pela implementação do agente-ladrão. O corpo (arquitetura) de ambos os agentes ocupa somente uma única célula do ambiente.
- O agente-poupador tem como objetivo guardar a maior quantidade de moedas possíveis, para isso ele terá a opção de guardá-las no banco. Evitando assim que o ladrão as roube. O agente-poupador sabe onde fica o banco (*Constantes.posicaoBanco*). Para isso ele terá um tempo predefinido que pode ser visto no canto superior à esquerda do ambiente. A cada movimento seu no ambiente, o tempo é reduzido em uma unidade. No ambiente existem algumas pastilhas “pastilhas do poder” que imunizam o poupador não deixando o agente-ladrão lhe roubar, porém essas pastilhas têm um custo. Para consegui-las o agente-poupador deve dar em troca 5 moedas. Ficando imune por 15 jogadas.
- O agente-ladrão também tem como objetivo encontrar moedas, porém ele só as consegue roubando do agente-poupador.
- Ambos os tipos de agentes dispõem de sensores de natureza visual, ou seja, suas percepções atuais (repassadas pelo ambiente à sua arquitetura) guardam informações visuais da parte do mundo à sua volta, dentro de um raio especificado, o qual, por sua vez, varia de um tipo de agente para outro.
- A Fig. 2 representa a visão atual dos agentes (visão 5 X 5). Deve-se notar que o agente (poupador ou ladrão) está representado pela célula de cor branca nessas figuras. Já as células em preto e em cinza representam as outras células que o agente pode perceber;

os quadros em cinza representam as posições para onde o agente pode se locomover. Além das percepções visuais, o agente-poupador recebe também dos seus sensores informações a respeito (i) quantidade de moedas em mãos e quantidade já depositadas; (ii) da coordenada (x,y) da posição atual que ocupa na matriz do labirinto; (iii) número de jogadas imunes. Além das percepções visuais, ele recebe também do ambiente informação a respeito de possíveis marcas olfativas (ferormônio) deixadas recentemente pelo agente-poupador e pelo ladrão no seu raio de visualização olfativa atual (ver Fig. 3).

0	1	2	3	4
5	6	7	8	9
10	11		12	13
14	15	16	17	18
19	20	21	22	23

Figura 2- Representação da visão do agente-poupador – ver possíveis valores na Tab. 1

0	1	2
3		4
5	6	7

Figura 3 - Representação do olfato dos agentes – ver possíveis valores na Tab. 2

Código	Significado
-2	Sem visão para o local
-1	Fora do ambiente (mundo exterior)
0	Célula vazia (sem nenhum agente)
1	Parede
3	Banco
4	Moeda
5	Pastinha do Poder
100	Poupador
200	Ladrão

Tabela 1 – Código das possíveis percepções visuais aparecendo nas células das Figs. 2

Código	Significado
0	Sem nenhuma marca
1	Com marca de cheiro gerada a uma unidade de tempo atrás
2	Com marca de cheiro gerada a duas unidades de tempo passadas
3	Com marca de cheiro gerada a três unidades de tempo passadas
4	Com marca de cheiro gerada a quatro unidades de tempo passadas
5	Com marca de cheiro gerada a cinco unidades de tempo passadas

Tabela 2 – Código das possíveis percepções olfativas aparecendo nas células da Fig. 3

- Com relação aos atuadores, ambos os tipos de agentes dispõem de membros (pernas) para se locomoverem no ambiente. A cada unidade de tempo, eles só podem se mover para uma célula adjacente ou subjacente, ou decidirem ficar parados na célula atual. Ambos os agentes não podem atravessar ou sobrepor às paredes; também não podem

ver por trás das paredes. Caso o agente decida se mover para uma célula inválida (p. ex., parede), o ambiente o manterá na célula atual. À medida que os agentes se movimentam pelo ambiente, eles deixam marcas, as quais permanecem ativas por apenas cinco unidades de tempo de simulação, após o que elas evaporam. Deve-se ressaltar que essas marcas são deixadas sempre, ou seja, já fazem parte dos atuadores de cada agente.

- O jogo chega a seu final, quando terminar o tempo. O vencedor será o agente que conseguir uma maior quantidade de moedas.
- Os grupos responsáveis pelo projeto do agente-poupador deverão implementar seu código via classe em Java que herde da classe abstrata ProgramaPoupador. Já os que projetarem o agente-ladrão deverão fazer o mesmo só que estendendo da classe abstrata ProgramaLadrao. Todos os métodos necessários para realizar as percepções (visuais ou não) e atuações (movimentação) dos agentes já estarão disponíveis no *framework* computacional (Fig. 5) como:

Para o Poupador:

```
sensor.getVisaoIdentificacao();  
sensor.getAmbienteOlfatoLadrao();  
sensor.getAmbienteOlfatoPoupador();  
sensor.getNumeroDeMoedas();  
sensor.getNumeroDeMoedasBanco();  
sensor.getNumeroJogadasImunes();  
sensor.getPosicao();
```

Para os Ladrão:

```
sensor.getVisaoIdentificacao();  
sensor.getAmbienteOlfatoLadrao();  
sensor.getAmbienteOlfatoPoupador();  
sensor.getNumeroDeMoedas();  
sensor.getPosicao();
```

Para ambos os agentes

- 0 - Ficar Parado() – o agente fica parado e não perde energia
- 1 - MoverCima() – move o agente uma posição acima da atual
- 2 - MoverBaixo() – move o agente uma posição abaixo da atual
- 3 - MoverDireita() – move o agente uma posição a direita da atual
- 4 - MoverEsquerda() – move o agente uma posição a esquerda da atual

Em ambas as classes abstratas deverá ser implementado o método ação.