

PROJETO AGENTES

Alunos:

David Silveira

Lucas Rosa

Novack Renato

Tema: 1 Modificar Agente

- O agente deverá ser capaz de limpar todo o ambiente independente do tamanho.
- Deverá ser explicado a estratégia das tomadas de decisão do agente.
- O ambiente deve conter obstáculos e o agente deverá ser capaz de contorná-los.

Estratégia utilizada

- Na aplicação do agente, o ambiente é do tipo **DETERMINÍSTICO**, ou seja, o próximo estado do ambiente é completamente determinado pelo estado atual e pela ação executada pelo agente.

PEAS

AGENTE	MEDIDA DESEMPENHO	AMBIENTE	ATUADORES	SENSORES
Robô Limpa de Sujeira	Limpar o ambiente e desviar dos obstáculos.	Piso	Rodinhas, Motor para locomoção, Compartimento para armazenar sujeira, sistema de sucção.	Câmera, sensores de presença.

Desviar do obstáculo

```
public void movimentar() {  
    if (this.pilhaMovimentos >= 4) {  
        return;  
    }  
    PosicaoXY proximoMovimento = retornarMovimento();  
    String valor = this.labirinto.retornarValorPosicaoLabirinto(proximoMovimento);  
  
    if (valor.equals("L") || valor.equals("*A*") || valor.equals("O")) {  
        proximoMovimento();  
        aumentarPilha();  
        movimentar();  
        if (valor.equals("O") && valor.equals("L")) {  
            movimentoObstaculo();  
            aumentarPilha();  
        }  
    } else {  
        this.labirinto.limpar();  
        this.posXY = proximoMovimento;  
    }  
}
```

Inserir obstáculo no labirinto

// Construir o labirinto// Neste metodo foi inserido a funcionalidade de criar o obstaculo dentro do labirinto utilizando
// O Random..

```
private void construirNovoLabirinto() {  
    labirinto = new String[this.tamanhoLabirinto][this.tamanhoLabirinto];  
    for (int i = 0; i < this.tamanhoLabirinto; i++) {  
        for (int j = 0; j < this.tamanhoLabirinto; j++) {  
            this.labirinto[i][j] = "S";  
            Random obstaculo = new Random();  
            int x = obstaculo.nextInt(2);  
            if(x==1) {  
                i = obstaculo.nextInt(this.getTamanhoLabirinto()-1);  
                j = obstaculo.nextInt(this.getTamanhoLabirinto()-1);  
                this.labirinto[i][j] = "O";  
            }  
        }  
    }  
}
```