

# Projeto Sobre Agentes

Tema: 4. Implementar um modelo de aprendizado

## Introdução

Para a implementação de um modelo de aprendizado de Agentes, que impede o agente de retornar para os espaços limpos dentro de um período de três segundos, foi necessário a utilização de *HashMap* e de programação paralela.

## Modelo de Aprendizado

Ao iniciar o programa o Labirinto e o Agente são criados, e o Agente é colocado em uma posição relativa ao tamanho do labirinto e nunca irá parar de se movimentar. O Agente limpa os espaços sujos e armazena em sua memória os locais limpos utilizando o *HashMap*, que armazena itens por índices, durante três segundos.

Ao tomar a decisão do próximo movimento, o agente verificar o local, caso esteja limpo e ainda não passou os três segundos de espera, ele irá para outra direção. Está tomada de decisão faz o uso do *HashMap*, que verifica se o próximo movimento ainda permanece armazenado.

Após o armazenamento do local que foi limpo, é iniciada uma *thread* para fazer a remoção da posição armazenada. Está *thread* recebe como nome a posição limpa e inicializa uma nova função que aguardar os três segundo e remove a posição do *HashMap*.

## PEAS

Agente	Medida de Desempenho	Ambiente	Atuadores	Sensores
Aspirador de pó	Realizar a limpeza sem retornar as posições em um período de 3s.	Local Sujo e Local Limpo	Realizar a movimentação, verificar se o ambiente está limpo, guardar e remover da memória a posição limpa.	Cima, baixo, esquerda e direita. L (limpo), S (sujo) e *A* (agente).

## Propriedades do Peas

- Parcialmente observável
- Determinístico
- Sequencial

- Dinâmico
- Contínuo
- Agente único

## **Atividades Executadas**

### **Grupo**

Carlos Emílio Terêncio de Oliveira Souza 53115

Gustavo Lopes dos Santos 53158

Lucas Vinicius Lima Braga de Amorim 54771

### **Gustavo e Lucas**

- Implementação do código.
- Documentação

### **Carlos**

- Slide

### **Repositório do Projeto**

<https://github.com/HannigUsT/Inteligencia-Artificial>