

# INTELIGÊNCIA ARTIFICIAL

## NAVEGAÇÃO DE ROBÔ PARA MANUTENÇÃO

### Descrição:

O trabalho consiste em implementar um sistema de navegação automática de um robô que realiza diferentes manutenções em unidades fabris. Para isto, deve-se utilizar o algoritmo de busca A\* para o cálculo do custo da rota.

O robô tem a informação da posição das fábricas a serem realizadas as manutenções, mas está desprovido de ferramentas para realizá-las. As ferramentas estão dispersas no ambiente e o robô deve capturá-las utilizando seu radar para poder se dirigir para a fábrica. Ou seja, o agente não tem a informação da localidade das ferramentas e só pode realizar a manutenção se possuir a ferramenta necessária.

Existem 5 tipos de ferramentas dispersas no ambiente:

- Bateria de carga elétrica;
- Braço de solda;
- Bomba de sucção;
- Dispositivo de refrigeração;
- Braço pneumático.

A posição inicial das ferramentas é desconhecida. O agente deve utilizar seu radar para localizá-las. O radar possui um alcance máximo de 4 regiões adjacentes em todas as direções. A Figura 1 ilustra o alcance máximo do radar considerando que o agente está localizado na posição marcada em vermelho.

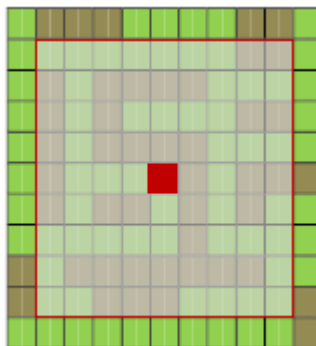


Figura 1: Alcance máximo do radar.

O total de cada tipo de ferramenta disponível no ambiente é o seguinte:

- 10 baterias de carga elétrica;
- 8 braços de solda;
- 6 bombas de sucção;
- 4 dispositivos de refrigeração;
- 2 braços pneumáticos.

Existem 5 tipos de fábricas que necessitam manutenção e que se relacionam com as ferramentas:

- Indústria de melhoramento genético de grãos com falta de energia elétrica nas incubadoras. Necessita de 2 baterias;
- Empresa de manutenção de cascos de embarcações. Necessita de 2 braços de solda;
- Indústria petrolífera com dutos entupidos. Necessita de 2 bombas;
- Fábrica de fundição com superaquecimento nas caldeiras. Necessita de 2 dispositivos de

- refrigeração;
- Indústria de vigas de aço com falta de braços mecânicos para moldagem. Necessita de 2 braços pneumáticos.

O ambiente por onde o robô vai navegar, representado através de uma matriz  $n \times n$ , é formado por diversos tipos de terrenos e em cada tipo de terreno o robô tem um grau de dificuldade diferente para andar. A Figura 1 mostra o ambiente a ser explorado pelo robô. Por exemplo, o robô consegue passar facilmente por um terreno sólido e plano, porém terá dificuldades para andar em um terreno montanhoso ou em um pântano.

Os tipos de terrenos que compõem o ambiente são (ver cores na Figura 2):

- Sólido e plano (verde) – Custo: 1
- Montanhoso (marrom) – Custo: 5
- Pântano (azul) – Custo: 10
- Árido (vermelho) – Custo: 15

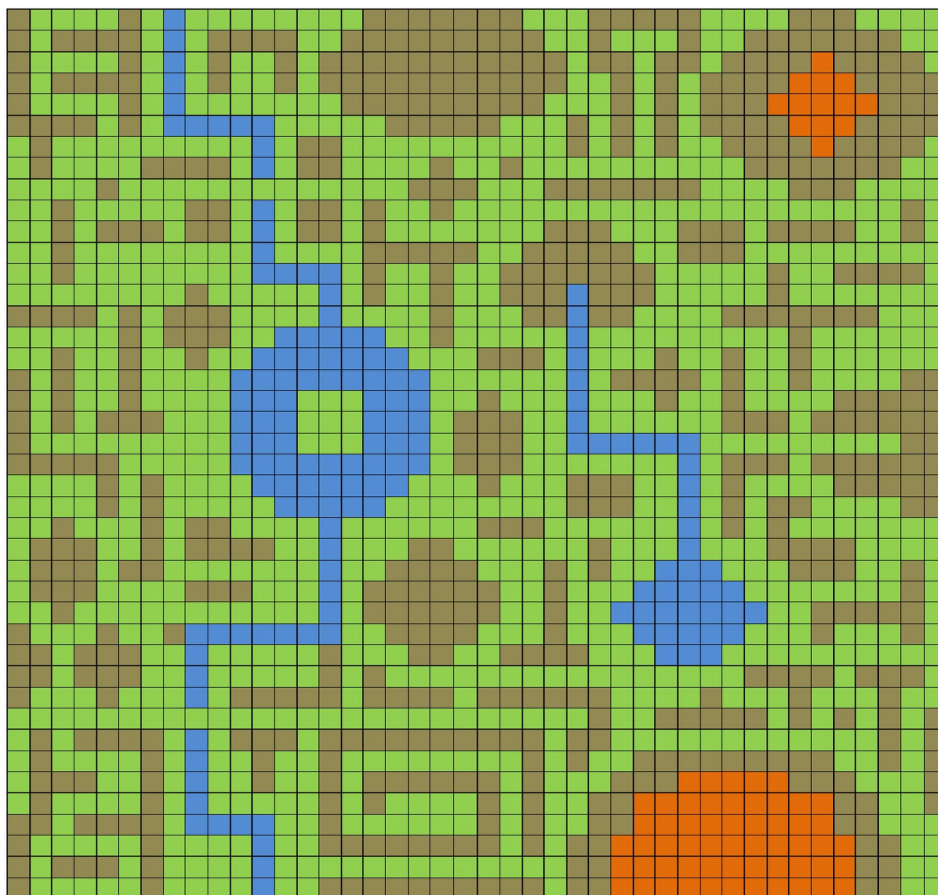


Figura 2: Ambiente a ser explorado pelo robô. Cores: verde – terreno sólido e plano; azul – pântano; marrom – montanhoso; vermelho – árido.

A melhor rota para consertar as 5 fábricas é a rota de menor custo levando em consideração o terreno.

#### Informações Adicionais:

- O robô pode andar somente na vertical e na horizontal;
- O ambiente deve ser representado por uma matriz  $42 \times 42$  (igual mostrado na Figura 2);
- O sistema deve permitir que o ambiente seja configurável (por arquivo de entrada ou pela

interface gráfica);

- A posição inicial do robô e das fábricas devem ser configuráveis (por arquivo de entrada ou pela interface gráfica). As fábricas podem estar em qualquer terreno do ambiente;
- O agente é capaz de identificar o tipo de ferramenta quando ela entra no alcance do radar e se ele já possui tal ferramenta;
- Caso mais de uma ferramenta apareça no radar, o robô deve calcular o melhor caminho e a melhor ordem para capturar as ferramentas, caso ele ainda não as possua;
- As ferramentas devem sempre estar em locais de grama. Ao sortear a posição inicial das ferramentas, o programa deve garantir que eles sempre estejam em regiões de grama;
- Deve existir uma maneira de visualizar os movimentos do agente, mesmo que a interface seja bem simples. Podendo até mesmo ser uma matriz desenhada e atualizada no console;
- O programa deve exibir o custo do caminho percorrido pelo agente enquanto ele se movimenta pelo mapa e também o custo final ao terminar a execução;
- O programa pode ser desenvolvido em qualquer linguagem;
- O trabalho individual.

### **Forma de avaliação:**

Será avaliado se:

- O trabalho atendeu a todos os requisitos especificados anteriormente;
- Os algoritmos foram implementados e aplicados de forma correta;
- O código foi devidamente organizado;
- O trabalho foi apresentado corretamente em sala de aula. O programa deve ser apresentado durante a aula por todos os membros do grupo. Se algum dos membros do grupo não comparecer ou não souber explicar nada sobre a implementação receberá nota zero.

### **Data de entrega:**

A definir. Apresentação em sala.

### **Importante:**

Os trabalhos completos de implementação (código e relatório) devem ser entregues **ATÉ** as datas do cronograma. A não entrega até estas datas atribuem valor **zero** ao trabalho.