

## Inteligência Artificial

2022/2023 – 2º Semestre

---

### PROJETO – PICKING, RECOLHA DE PRODUTOS EM ARMAZÉNS

---

#### 1. Descrição do problema

O objetivo deste projeto é o desenvolvimento de uma aplicação que permita otimizar a recolha de produtos das prateleiras de um armazém, um processo conhecido por *picking*. O objetivo do sistema consiste em distribuir os *picks* (produtos) pelos agentes responsáveis pela recolha, bem como definir a ordem pela qual cada agente deve recolher os produtos que lhe foram atribuídos de modo a minimizar o tempo de entrega do último produto a ser entregue no ponto de entrega. Pretende-se também minimizar a distância total percorrida pelos agentes e o número de colisões entre os agentes.

O programa deve começar por permitir que o utilizador escolha o problema a resolver. O problema é guardado num ficheiro de texto. A figura abaixo mostra um exemplo de um destes ficheiros.

```
19, 21
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
2 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
1 0 2 1 0 1 1 0 2 1 0 1 1 0 1 1 0 1 1 0 1
1 0 1 1 0 1 1 0 1 1 0 2 1 4 1 2 0 1 1 0 1
1 0 2 1 0 1 2 0 1 1 0 1 1 0 1 1 0 1 1 0 1
1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
1 0 1 1 0 1 1 0 1 1 0 2 1 0 1 1 0 1 1 0 1
1 4 1 1 0 1 2 0 1 2 0 1 1 0 1 1 0 1 2 0 1
1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
1 0 1 1 0 1 1 0 1 1 0 1 1 0 2 1 0 1 1 0 1
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
1 0 1 1 0 1 1 0 1 2 0 1 1 0 1 1 0 1 1 0 1
1 0 1 1 0 1 1 0 1 1 4 2 2 0 2 1 0 2 1 0 2
1 0 1 2 0 1 1 0 1 1 0 1 1 0 1 2 0 1 1 0 1
1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1 1 0 1
0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 0 0 0 0 0
```

A primeira linha indica o comprimento e a largura do armazém; Segue-se uma matriz que descreve a estrutura do armazém, a posição dos produtos, a posição do ponto de recolha e a posição inicial dos agentes. Os valores têm o seguinte significado: 0 indica um espaço livre por onde os agentes podem circular; 1 representa uma prateleira vazia; 2 representa uma prateleira com um produto a ser recolhido; 3 representa o ponto de entrega; 4 representa um agente.

Uma vez distribuídos os produtos pelos agentes, cada agente deve dirigir-se ao primeiro produto a ser por si recolhido, depois ao segundo e assim por diante até ao último. Depois de recolher o último produto, o agente deve dirigir-se para o ponto de entrega. Todos os percursos realizados pelo agente devem ser percursos ótimos. Em cada passo de um percurso, o agente pode mover-se para uma das células adjacentes Norte, Sul, Este, Oeste, desde que a célula exista ou não esteja ocupada por uma prateleira. Considera-se que o agente recolhe um produto quando é a vez desse produto ser recolhido e o agente passa na célula adjacente à célula onde o produto está localizado.

Considera-se que dois agentes colidem se tentarem ocupar a mesma célula no mesmo momento. Note-se que neste projeto não se pretende que a aplicação procure encontrar um caminho alternativo quando ocorre uma colisão entre dois agentes. Pretende-se “apenas” que as colisões sejam detetadas e que as soluções sejam penalizadas consoante o número de colisões que ocorrerem. O objetivo é que o processo evolutivo favoreça soluções sem colisões na tentativa de que a melhor solução encontrada não implique a ocorrência de colisões.

## 2. Trabalho a realizar

O objetivo principal deste projeto é o desenvolvimento de uma aplicação que utilize algoritmos genéticos e o algoritmo de procura A\* para resolver problemas como o descrito acima. Cada solução do algoritmo genético deve representar uma possível distribuição de produtos pelos agentes, bem como a ordem pela qual o agente deve recolher produtos que lhe foram atribuídos. O algoritmo A\* deve ser utilizado para calcular o caminho entre a posição inicial de cada agente e o primeiro produto a ser por si recolhido, entre as posições dos sucessivos produtos a serem recolhidos e entre o último produto e o ponto de entrega.

A aplicação a ser desenvolvida deve permitir visualizar a simulação da melhor solução encontrada.

O projeto fornecido com este enunciado é baseado nos projetos desenvolvidos nas aulas e já contém alguns aspetos implementados, como sejam a leitura dos ficheiros que contêm os problemas (datasets) e a GUI da aplicação, e outros parcialmente implementados, como sejam a representação do problema, o estado, o algoritmo A\* e o algoritmo genético.

O projeto consiste na realização das tarefas seguintes:

1. Desenvolvimento da classe que permite representar um estado do problema (classe WarehouseState);

2. Desenvolvimento da classe que representa o problema, a ser utilizada pelo algoritmo A\* (classe WarehouseProblemSearch);
3. Desenvolvimento de uma heurística para o problema, a ser utilizada com o algoritmo A\* (classe HeuristicWarehouse);
4. Desenvolvimento de classe que representa o problema, a ser utilizada pelo algoritmo genético (classe WarehouseProblemGA);
5. Desenvolvimento da classe que permita representar um indivíduo do algoritmo genético (classe WarehouseIndividual);
6. Desenvolvimento de dois operadores de recombinação adequados ao problema e à representação utilizada para os indivíduos (classes Recombination2 e Recombination3);  
*Nota:* já é disponibilizado um operador de recombinação (classe RecombinationPartialMapped).
7. Desenvolvimento de dois operadores de mutação adequados ao problema e à representação utilizada para os indivíduos (classes Mutation2 e Mutation3);  
*Nota:* já é disponibilizado um operador de mutação (classe MutationInsert).
8. Realização de testes que permitam estudar os seguintes aspetos:
  - O efeito da variação do tamanho da população e do número de gerações;
  - Desempenho relativo dos diferentes operadores genéticos;
  - O efeito da variação das probabilidades dos operadores genéticos utilizados;
  - O efeito da variação do tamanho do torneio.

Do relatório do projeto deve constar:

- A descrição do estado;
- A descrição da heurística;
- A descrição da representação dos indivíduos utilizada no algoritmo genético;
- A descrição da função de avaliação (função de *fitness*) utilizada;
- A descrição do método de criação da população inicial;
- A descrição dos operadores genéticos desenvolvidos;
- A apresentação e discussão dos resultados obtidos nos testes realizados;
- Outros aspetos considerados relevantes para uma boa compreensão e avaliação do trabalho realizado.

Alguns dos fatores com mais importância na avaliação do relatório são:

- Clareza na descrição dos componentes da aplicação;
- A forma como os resultados dos testes são compilados e a clareza com que são apresentados (a utilização, mas não só, de tabelas e/ou gráficos pode ajudar);
- O significado estatístico dos resultados (que depende do número de *runs* realizados);
- A análise e discussão dos resultados.

### 3. Cotações

20% - Tarefas 1 e 2

5% - Tarefa 3

30% - Tarefas 4 e 5

10% - Tarefas 6 e 7

25% - Tarefa 8 e Relatório (devem ser entregues todos os ficheiros das experiências)

10% - Extras

Extras sugeridos:

- Penalizar as colisões pelo tempo que demorariam a ser evitadas e não pelo seu número. Pode considerar-se, por exemplo, que para evitar uma colisão, um dos agentes tenha que esperar junto a um cruzamento para o outro agente passar;
- Implementação de outras metaheurísticas;
- Utilizar um algoritmo de *clustering* para atribuir os *picks* aos agentes e utilizar o algoritmo genético apenas para definir a ordem pela qual os agentes recolhem os produtos.

### 4. Prazos, datas, regras e instruções

1. Data limite de entrega do projeto: **27 de junho de 2023**.
2. Data das defesas: **3 de julho de 2023**.
3. O projeto é realizado em grupos de 2 estudantes. Não são aceites projetos realizados por grupos com mais de 2 elementos. Os estudantes que pretendam realizar o projeto individualmente devem solicitá-lo, por escrito, ao docente responsável pela UC. Apenas em casos bem fundamentados serão autorizados projetos realizados individualmente.
4. O relatório deve ser realizado utilizando o modelo disponibilizado na secção *Projeto* do sítio da UC no Moodle.
5. O projeto deve ser entregue sob a forma de um arquivo zip, rar ou 7z que contenha todos os elementos do projeto, incluindo o relatório. O nome do arquivo deve ter o formato *IA\_Projeto\_#1\_#2.(zip/rar/7z)*, onde #1 e #2 devem ser substituídos pelos números de estudante dos elementos do grupo. O relatório deve ser entregue em formato pdf e o seu nome deve ter a mesma estrutura do arquivo mas com extensão pdf.