

# Projeto de Fundamentos de Programação

## Manual do utilizador

GRUPO N° 66

103847	Guilherme Fernandes guilhermegrancho@tecnico.ulisboa.pt
103332	Gonçalo Drago goncalo.drago@tecnico.ulisboa.pt

Licenciatura em Engenharia Mecânica

Licenciatura em Engenharia e Arquitetura Naval

Lisboa, junho de 2022

# Manual do utilizador

## Índice

1.	Sumário.....	3
2.	Manual do utilizador.....	4
2.1.	Descrição do programa .....	4
2.2.	Guia de utilização.....	5
2.3.	Limitações do programa.....	6

# Manual do utilizador



## 1. Sumário

Este programa baseia-se na criação de um robô que se movimenta numa área à procura dos vários objetivos selecionados pelo utilizador ou selecionados por um ficheiro. Nesta área é gerado aleatoriamente ou através de um ficheiro vários obstáculos aos quais o robô tem de ter a capacidade de contorná-los e sem se sobrepor a eles caso seja necessário no caminho até ao obstáculo. O programa tem a capacidade de fazer com que o robô contorne os obstáculos e reconheça os objetivos de forma a encontrá-los e depois de ter encontrado todos volta à base. Este programa cumpre com todos os objetivos das quatro implementações que foram pedidos que são falados mais aprofundadamente na descrição do programa. Desta forma não há nada que não esteja implementado ou que não esteja a funcionar. O tempo de execução do projeto foi aproximadamente 150 horas.

# Manual do utilizador

## 2. Manual do utilizador

### 2.1. Descrição do programa

Neste programa foi-nos pedido que desenvolvêssemos quatro diferentes implementações.

A primeira implementação tem como objetivo criar uma janela onde vai ser a área de navegação do robô. Nesta área vai ser criado vários obstáculos que vão ser espalhados aleatoriamente (no nosso caso escolheu-se com a forma de círculos, todos com a mesma dimensão). O utilizador vai selecionar vários pontos na janela de uma só vez que vão ser os objetivos. O robô vai ter que procurar os objetivos, sabendo que quando encontra um obstáculo choca contra ele não se sobrepondo, e regressar à base.

Os objetivos da segunda implementação são iguais aos da primeira, no entanto os obstáculos vão ter várias formas, cores (círculos e quadrados) e várias dimensões escolhidos de forma aleatória. O robô também vai ter uma nova implementação que é a bateria. Esta bateria vai reduzindo o seu valor à medida que o robô navega na área e quando a bateria está a meio ele volta a base para carregar. E depois prossegue na recolha dos objetivos.

Na terceira implementação as cores, as dimensões e formas dos obstáculos são definidas num ficheiro chamado de “Ambiente.txt” onde o programa deve ter capacidade de processar os dados e criar os obstáculos correspondentes.

A quarta implementação é a que tem mais obstáculos e existem dois modos diferentes. Num modo os dados dos objetivos estão definidos num ficheiro no outro os dados dos objetivos são introduzidos através de cliques do rato.

O nosso programa cumpre com todos os objetivos que foram pedidos.

Ao realizar o projeto percebemos que este programa tem uma área vasta de aplicação desde que o objetivo seja localizar um objetivo e fazer a recolha dele. Como por exemplo um veículo robótico com fins militares onde o seu objetivo é localizar minas e recolhê-las.

# Manual do utilizador

## 2.2. Guia de utilização

5

Ao correr o ficheiro Main.py aparece um menu ao qual tem três opções play, help e exit. Se o utilizador clicar na opção exit o programa é fechado. Se o utilizador clicar na opção help o programa abre uma janela informativa. Ao utilizador clicar na opção play o programa abre uma janela com quatro opções diferentes que correspondem a quatro modos de jogo diferentes. O utilizador ao selecionar um dos quatro modos o programa abre uma janela onde se procede o jogo. Nesta janela é apresentado a área de navegação do robô com duas bases. Uma no canto superior esquerdo e outra no canto inferior direito onde o robô inicialmente se encontra posicionado. Nesta área também existem obstáculos que são gerados aleatoriamente. Ao clicar numa zona em específico na área de navegação do robô será posto um círculo vermelho chamado de objetivo. Estes objetivos são espalhados pelo mapa de acordo com o gosto do utilizador. Depois de ter selecionado o último objetivo o utilizador deve clicar no robô para que este comece a recolher os objetivos. Depois de o robô ter recolhido o último objetivo e voltado à base o programa fecha a janela e abre a janela anterior onde o utilizador pode selecionar novamente o modo de jogo. Caso o utilizador queira sair a meio basta clicar na tecla “e”.

Ao clicar na quarta implementação o programa abre uma janela onde tem duas opções para duas diferentes maneiras de introduzir os objetivos. Uma é através de um ficheiro onde já estão inseridos esses dados e outra é através de cliques no rato. Para alterar os dados deste ficheiro, cujo nome é “Limpeza.txt”, deve escrever as coordenadas no formato de x y separados por um espaço, por exemplo: “62.5 62.5”. É recomendável não introduzir mais do que uma casa decimal. Saber também que cada parágrafo (linha) no ficheiro corresponde a um objetivo diferente, por exemplo se quiser três diferentes objetivos o ficheiro deve ter três linhas.

Na terceira implementação é possível mudar os dados dos obstáculos e o tamanho da janela através do ficheiro “Ambiente.txt”. Para o utilizador criar um obstáculo neste ficheiro tem de escrever no formato de cor e depois forma geométrica, separados por um espaço. Existem duas cores e duas formas geométricas disponíveis, verde ou cinzento e círculo ou retângulo. Caso o utilizador escolha a cor verde deve escrever “Grass” ou “Bush” caso a cor escolhida seja cinzenta o utilizador deve escrever “Stone”. Quanto às formas geométricas, caso o utilizador queira um círculo deve selecionar um ponto com coordenadas (x,y) e um raio, separados por uma vírgula. Desta forma deve escrever por exemplo “Circle(Point(50,50),4)”. Caso o utilizador queira um retângulo deve selecionar dois pontos com coordenadas (x,y) separados por uma vírgula. Desta forma deve escrever por exemplo “Rectangle(Point(60,80),Point(65,85))”. Caso o utilizador escreva tudo em minúsculas ou tudo em maiúsculas ou alternadamente entre minúsculas e maiúsculas o programa consegue mesmo assim correr. E caso o utilizador não escreva parênteses e vírgulas ou estejam parenteses e

# Manual do utilizador

vírgulas mal postos o programa consegue verificar o pretendido desde que tudo esteja separado com um espaço. Por exemplo: “STONE),(CiRcLe ,point, 50() 50 4,,”.

Quanto ao tamanho da janela, para o utilizador mudar as dimensões da janela deve inserir os valores no formato de (x,y) separados por um espaço, como por exemplo “500 500”.

Na segunda implementação quando está a regressar para base ou está na base a carregar, para o bom funcionamento é necessário clicar duas vezes no “e” para sair do programa caso contrário o programa deixa de funcionar corretamente.

Caso o utilizador queira logo sair do jogo sem selecionar nenhum objetivo, é necessário que clique na tecla “e” e de seguida no robô. Exceto na implementação quatro onde os pontos já são carregados pelo ficheiro.

O programa produz o ângulo e a distância que o robô deve percorrer em conjunto com a informação sobre se o robô colidiu ou não com algum obstáculo, com a lista dos objetivos e de obstáculos falados anteriormente e com o caminho para os objetivos e para a base dada a posição do robô.

## 2.3. Limitações do programa

Visto que o robô está sempre a rodar para a esquerda quando choca num obstáculo o robô vai sempre desviar-se pela esquerda, ou seja, mesmo que o caminho para o objetivo seja menor ao contornar o obstáculo pela direita o robô vai contornar o obstáculo pela direita.

O robô vai à procura dos objetivos por ordem e não pelo que está mais próximo.

Ter em atenção que quando se clica na tecla “e” esta não pode estar em maiúscula, caso contrário o programa não reconhece a dada tecla.