

Tecnologias e Sistemas de Informação para a Web

CURSO

2022/2023	2023/09/07	14h30	1h45
-----------	------------	-------	------

ANO LETIVO

DATA

HORA

DURAÇÃO

Algoritmia e Estruturas de Dados

1º ano

UNIDADE CURRICULAR

ANO

Mário Paulo Teixeira Pinto

Especial

DOCENTE

ÉPOCA

Observações:

- O Teste é individual e de consulta. Resolva os exercícios recorrendo à linguagem Python
- **Crie uma pasta no Ambiente de Trabalho com o seu número-nome. Ex: 40200999 – Maria Zulmira**
- **Guarde os exercícios resolvidos nessa pasta**
- **No final do teste compacte a pasta e submeta-a no moodle.**

### I (25%)

Crie um programa (**Exercicio1.py**) que permita gerar um número aleatório entre [1-100] e o apresente na Console.

Em seguida o seu programa deve perguntar ao utilizador: “*Deseja gerar novo número (S/N)?*”.

- Se responder Não: termina o programa, apresentando a lista de todos os números gerados, ordenados de forma decrescente.
- Se responder sim: deve gerar novo número aleatório **entre o último número gerado e 100**

Notas:

- Deve assegurar-se que os números aleatórios **nunca** se repetem
- Caso o número gerado aleatoriamente seja 100, deve terminar o programa apresentando a lista de números gerados, ordenados de forma decrescente

### II (25%)

Crie um programa (**Exercicio2.py**) em ambiente de console que permita pesquisar nomes numa lista (listName).

Comece por criar uma lista no seu programa com um conjunto de nomes, como no exemplo abaixo:

```
listName = ["Alexandre Dias", "Carla Dias", "Fábio Carvalho", "Pedro Fonseca", "Alexandre Diogo"]
```

Em seguida o seu programa deve pedir ao utilizador a introdução de um nome de pesquisa (nameFind) e invocar a função **searchName(listName, nameFind)**. Esta função, que vai codificar, recebe como argumentos a lista de nomes e o nome a pesquisar, e deve devolver quantos nome da lista incluem o objeto de pesquisa.

Exemplo:

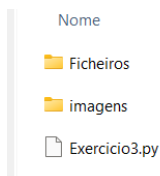
nameFind = "Alexandre" -> A função serachName deve devolver 2.

nameFind = "Dias" -> A função serachName deve devolver 2.

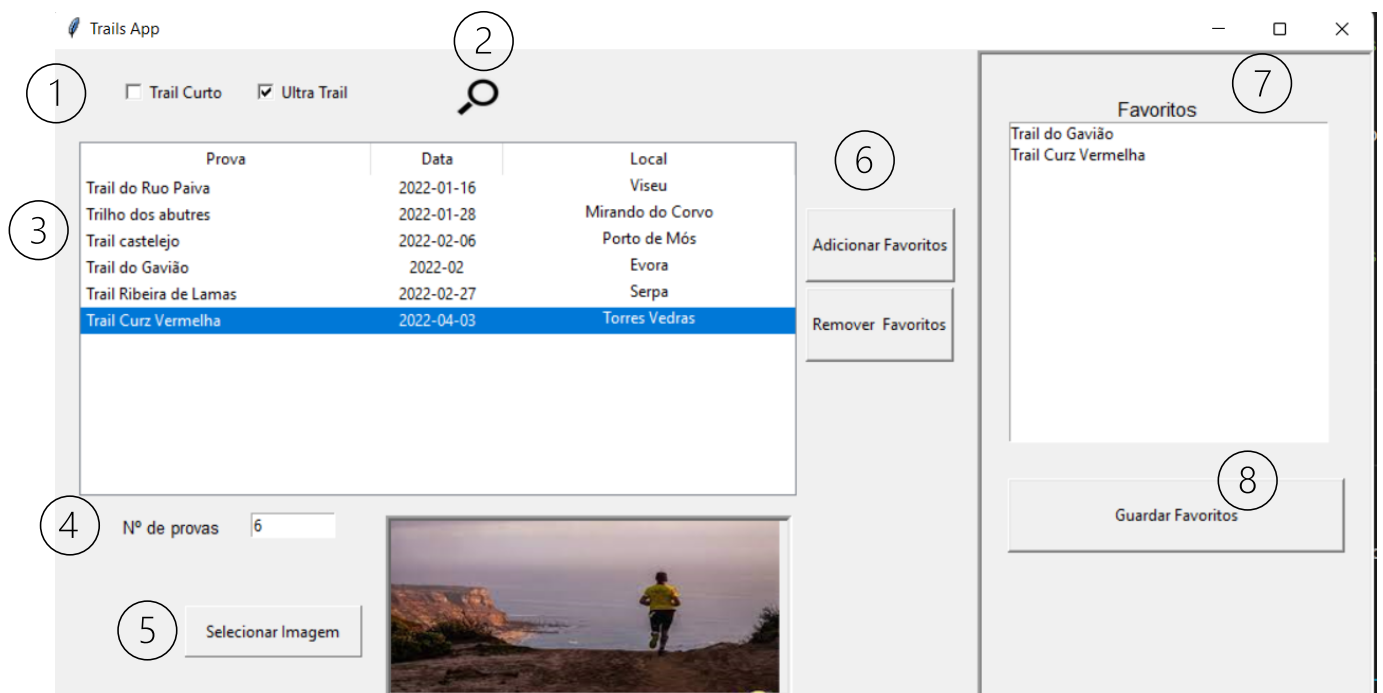
nameFind = "Fonseca" -> A função serachName deve devolver 1.

### III (50%)

Descarregue no Moodle o ficheiro **Epoca Especial - EX3**. Descompacte o ficheiro de forma a obter a seguinte estrutura (dentro da pasta do exercício):



Implemente uma pequena aplicação (**Exercicio3.py**) semelhante à abaixo apresentada:



Algumas notas:

Window com 1000x500, título: "Trails App".

- 1- Checkbuttons, o primeiro deve estar ativo por defeito. O utilizador pode ativar um, ou dois, ou nenhum deles;
- 2- Button com imagem *pesquisar.png*, width=35, height=35;

As clicar no button:

- Se checkbox *Trail Curto* está ativo, deve ler o conteúdo do ficheiro *trails.txt* para a treeview;
- Se checkbox *Ultra Trail* está ativo, deve ler o conteúdo do ficheiro *ultratrails.txt* para a Treeview;
- Se os dois checkbox estão ativos, a treeview deve incluir os dados dos dois ficheiros

- 3- Treeview: 1ª coluna com alinhamento à esquerda, 2ª e 3ª coluna com alinhamento centrado.
- 4- Nº de provas visíveis na treeview
- 5- Selecionar uma das imagens existentes na pasta imagens, a visualizar no componente canvas
- 6- Adicionar a prova selecionada (na treeview) à lista de favoritos (é uma ListBox);  
Remover a linha selecionada (na listbox) da lista de favoritos
- 7- Label com texto Favoritos, com tipo de letra *Helvética*, tamanho 11. ListBox que contém os favoritos adicionados.
- 8- Guardar a lista no ficheiro favoritos.txt