

 <b>POLITÉCNICO DE LEIRIA</b> <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	TRABALHO PRÁTICO	<b>DEI</b>	Departamento Engenharia Informática
	PROGRAMAÇÃO		

**CURSOS:** Engenharia Mecânica (D+PL)

**ANO LETIVO:** 2022/2023

**SEMESTRE:** 1.º

**DATA:** 16-11-2022

## Trabalho Prático de Programação para o Curso de Engenharia Mecânica

### – *Análise de Testes de Escolha Múltipla* –

#### 1 - Enquadramento

O trabalho prático consiste no desenvolvimento de um programa informático codificado na linguagem de programação Python (versão 3) para a análise de resultados de testes de escolha múltipla.

Os testes de escolha múltipla são construídos a partir do programa executável `geraTeste.exe`, que se encontra no ficheiro `geraTeste.rar` e o qual é uma implementação codificada em Python que permite construir um teste com N perguntas de escolha múltipla, onde cada pergunta pode ter entre duas a quatro opções de escolha, sendo que apenas uma está correta. As N perguntas do teste encontram-se desordenadas e após a construção do teste, toda a informação é armazenada num ficheiro em formato TXT.

A Figura 1 ilustra a construção de um teste com 20 perguntas de escolha múltipla, o qual vai ser armazenado no ficheiro `testeEM.txt` e usando o programa executável `geraTeste.exe`.

```

Programa Gerador de Testes
-----
O programa gera ficheiros em formato
de texto com a informação relativa a
um teste de escolha múltipla
-----

Estrada de Dados
-----
Indique o Nome do Ficheiro TXT: testeEM.txt
Indique o Número de Perguntas: 20_

```

**Figura 1** Construção de um teste com 20 perguntas.

Cada uma das N perguntas de um teste de escolha múltipla é composta pelos seguintes elementos:

- A cadeia de caracteres --> Pergunta: <-- identifica o início da pergunta.
- O número N da linha seguinte identifica o número da pergunta.
- A cadeia de caracteres Texto da Pergunta N identifica o texto da pergunta N.
- O número M da linha seguinte identifica a opção correta da pergunta N.
- A cadeia de caracteres Resposta: ..... identifica o texto da opção de resposta da pergunta N.

A Figura 2 ilustra um exemplo de um teste de escolha múltipla em ficheiro.

```
--> Pergunta: <--
52
Qual dos seguintes tipos de estruturas de dados é mutável?
2
Cadeias de caracteres
Listas
Tuplos
Sets
--> Pergunta: <--
37
Texto da Pergunta 37
2
Resposta: ..... B
Resposta: ..... A
--> Pergunta: <--
```

Figura 2 Primeiras perguntas de um teste

## 2 – Características do Programa Informático

A execução do programa informático inicia com a apresentação ao utilizador do menu com as funcionalidades que passaremos a descrever e vai sendo executado até que o utilizador indique que pretende concluir a sua execução.

```
1 - Construir Respostas
2 - Analisar Resultados
0 - SAIR
```

### 2.1 – Funcionalidade “Construir Respostas”

Esta funcionalidade permite que o utilizador construa as respostas de um teste. A construção das respostas é efetuada de maneira aleatória e com os seguintes requisitos:

- O utilizador começa por indicar o nome do ficheiro em formato TXT que pretende abrir e no qual se encontra o teste de escolha múltipla para o qual vai construir as respetivas respostas.

Observação: a informação existente no ficheiro em formato TXT tem obrigatoriamente que ser armazenada em uma estrutura de dados do tipo dicionário.

- O número de respostas do teste do item anterior é obtido a partir de um ficheiro em formato CSV (ficheiro do tipo Excel com os valores separados por vírgulas), construído por cada grupo de trabalho e com duas colunas: a primeira coluna tem o Número de estudante com o formato AAAABBBB, onde AAAA é o ano letivo de ingresso e BBBB é o número da secretaria; a segunda coluna tem o Nome e o Apelido do(a) estudante.

Observações: podem existir estudantes com o mesmo Nome e Apelido, no entanto, têm Números de estudante diferentes; o nome do ficheiro em formato CSV a usar para a construção das respostas é indicado pelo utilizador.

- Combinando as informações dos ficheiros dos itens anteriores, para cada estudante são construídas as respostas do teste de escolha múltipla.

Observações: as respostas são construídas pela mesma ordem com que as perguntas aparecem no teste; para cada pergunta, a resposta é construída de maneira aleatória dentro das opções possíveis e o valor 0 identifica pergunta não respondida.

- Concluído o item anterior, toda a informação é armazenada em um ficheiro em formato CSV cuja cadeia de caracteres terminal é \_RES.CSV.

## 2.2 – Funcionalidade “Analisar Resultados”

Esta funcionalidade permite que o utilizador analise os resultados de um teste de escolha múltipla em relação às respostas construídas. Para isso, o utilizador tem que indicar qual é o teste de escolha múltipla que deseja analisar (ficheiro em formato TXT) e qual é o conjunto de respostas que deseja analisar (ficheiro em formato CSV com a cadeia de caracteres terminal \_RES.CSV).

A análise dos resultados apresenta itens obrigatórios e itens opcionais e quanto aos itens opcionais há que escolher dois. As opções implementadas deverão ser apresentadas ao utilizador em forma de menu (claramente identificadas):

- Itens obrigatórios:
  - 1) Listar a classificação final (entre 0,0 e 20,0 valores) de todos os estudantes.

Sugestão: usar o formato NúmeroEstudante - NomeApelido - Classificação.

A classificação obtida por um estudante é calculada somando a classificação das perguntas com duas opções de resposta com o valor de 1; a classificação das perguntas com três opções de resposta com o valor 2, e a classificação das perguntas com quatro opções de resposta com o valor de 4; as perguntas erradas descontam  $1/M$  % do valor da pergunta onde M indica o número de opções de resposta. A cotação de 0,0 a 20,0 é a que resulta da conversão desse total obtido.

- 2) Listar a informação de um estudante.

Sugestão: usar o formato:

```
NúmeroEstudante - NomeApelido
NúmeroRespostasCorretas:
NúmeroRespostasErradas:
NúmeroRespostasNãoRespondidas:
ClassificaçãoDoTeste:
ClassificaçãoFinal:
```

- 3) Pesquisar e apresentar o número da pergunta com mais respostas corretas.
- 4) Pesquisar e apresentar o número de respostas corretas, o número de respostas erradas e o número de respostas não respondidas de uma pergunta do teste indicada pelo utilizador.
- 5) Apresentar o gráfico de barras que no eixo das abcissas indica o número de opções de cada uma das perguntas e no eixo das ordenadas indica o número de perguntas existentes no teste e o número de respostas corretas nesse tipo de pergunta.
- 6) Apresentar o gráfico circular com a percentagem de estudantes aprovados e reprovados. Na mesma figura há que apresentar o gráfico de barras com as classificações obtidas pelos estudantes, divididas nas classes [0,7], [8,9], [10,12], [13,15], [16,17], [18,20].

- Itens opcionais:

- 1) Listar toda a informação relativa a uma pergunta do teste.
- 2) Listar o número de estudante dos estudantes que não obtiveram a classificação mínima exigida, sabendo que esta classificação é de 8,0 valores.
- 3) Pesquisar e apresentar o número da pergunta com mais respostas não respondidas.

- 4) Pesquisar e apresentar o número de perguntas do teste cuja opção correta é a indicada pelo utilizador.
- 5) Apresentar o gráfico de barras que no eixo das abcissas indica cada uma das opções de uma pergunta do teste e no eixo das ordenadas indica o número de perguntas em que essa opção é a correta.
- 6) Apresentar os gráficos circulares que para cada tipo de pergunta indica o número de perguntas cuja resposta correta é cada uma das opções.

### **2.3 – Funcionalidade “SAIR”**

Esta funcionalidade permite que o utilizador termine a execução do programa informático. Antes de “SAIR” do programa informático, há que perguntar ao utilizador se pretende mesmo terminar a execução do programa informático.

### **3 – Regras a Cumprir**

O trabalho prático apresenta um conjunto de regras a cumprir:

1. O trabalho prático deve ser realizado em grupos de dois estudantes.
2. A entrega do trabalho prático tem obrigatoriamente que incluir os seguintes elementos, em formato eletrónico:
  - Uma descrição esquemática (desenho) das estruturas de dados usadas no programa (ficheiro em formato PDF).
  - O algoritmo em pseudocódigo (máximo 5 páginas A4, ficheiro em formato PDF).
  - A listagem de todas as codificações na linguagem de programação Python (ficheiros em formato PY).
3. A identificação dos autores do trabalho prático deverá incluir, em cada um dos elementos a entregar, o Nome Completo, o Número de estudante e o Endereço de Correio Eletrónico. Em caso de falta de identificação, o trabalho prático não será avaliado.
4. A codificação do trabalho prático tem que estar estruturada por módulos e o programa principal tem que ter o menor número de instruções possível.

5. A codificação do trabalho prático tem obrigatoriamente que poder ser executada nos computadores da escola, sem haver a necessidade de recorrer à instalação de bibliotecas/módulos adicionais da linguagem de programação Python.
6. Na codificação do trabalho prático, cada função tem que ser documentada relativamente à sua funcionalidade e aos seus parâmetros entrada/saída. Exemplo:

```
def calculaNota(respostas)
    """
        Descrição: calcula a nota final de um teste de escolha múltipla
                    conhecidas as suas "respostas"
        Entrada: "respostas" é a estrutura de dados com as respostas
                  do teste
        Saída: "nota" é a nota final obtida no teste de escolha múltipla

        .
        .
        .
    """
    return nota #fim da função calculaNota
```

7. A codificação do trabalho prático tem que garantir execuções eficientes e portanto, antes do desenvolvimento dos elementos na linguagem de programação Python há que realizar uma análise sobre as estruturas de dados a utilizar e definir o algoritmo em pseudocódigo que sai ver codificado.
  8. Os aspetos que se considerem omissos no enunciado do trabalho prático devem ser resolvidos pelo grupo de trabalho e têm que ser descritos no documento do pseudocódigo, referindo o que se considerou omissos e qual a solução codificada.
  9. Todos os elementos do trabalho prático em formato digital são submetidos na plataforma Moodle, no link existente para esse efeito e na data e hora definida pelo calendário de avaliação.
- Observação: o link da plataforma Moodle identificará o Regime e o Nome do Docente que vai avaliar o trabalho prático.
10. O trabalho prático é avaliado em prova oral obrigatória.

#### 4 – Avaliação do Trabalho Prático

Para que o trabalho prático possa ser avaliado é obrigatório:

- Apresentar a estrutura de dados implementada.
- Apresentar o algoritmo e o pseudocódigo.

- Apresentar todas as codificações e elas têm que poder ser executadas nos computadores das salas de aula.

O trabalho prático é avaliado segundo os critérios que se encontram na Tabela 1 e a classificação final é dada por,

$$\text{Classificação Final} = \text{Classificação Trabalho Prático} * \text{Percentagem da Defesa}$$

onde **Percentagem da Defesa** varia entre 0 % e 100 %, em função da segurança do estudante na defesa do trabalho prático.

**Tabela 1:** Critérios de avaliação do trabalho prático.

Elementos Gerais do Trabalho Prático		40,0 %
A	Estrutura de Dados	15,0 %
B	Algoritmo e Pseudocódigo	10,0 %
C	Estrutura Geral do Programa Informático: Programa principal reduzido e correta estruturação em módulos Proteção de dados (mensagens de erro, ...) e de cálculos Uso adequado de menus e a sua correta gestão Uso adequado de comentários e de nomes sugestivos para variáveis, funções, ...	15,0 %
Codificação do Trabalho Prático		60,0 %
D	Utilização de Módulos de Python e de Módulos Próprios	7,5 %
E	Utilização de Funções	7,5 %
F	Utilização de Estruturas de Decisão	7,5 %
G	Utilização de Estruturas de Repetição	7,5 %
H	Utilização de Estruturas de Dados	10,0 %
I	Utilização de Ficheiros	10,0 %
J	Utilização de Gráficos	10,0 %