

# **Licenciatura em Engenharia Informática**

## **Estrutura de Dados**

**2019/2020**

### **Relatório**

#### **Ficha de Trabalho Nº2 - Ficheiros**

**Pedro Sousa 2019140586**

**Stéphane Quental 2019131493**

**07/04/2020**

## Índice

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>3</b>
<b>2</b>	<b>TRABALHO REALIZADO .....</b>	<b>4</b>
2.1	PERGUNTA 1 .....	4
2.2	PERGUNTA 2 .....	5
2.3	PERGUNTA 3 .....	6
2.4	PERGUNTA 4 .....	6
2.5	PERGUNTA 5 .....	7
2.6	PERGUNTA 6 .....	8
2.7	PERGUNTA 7 .....	9
2.8	PERGUNTA 8 .....	10
2.9	PERGUNTA 9 .....	11
<b>3</b>	<b>CONCLUSÃO .....</b>	<b>13</b>
<b>4</b>	<b>REFERÊNCIAS .....</b>	<b>14</b>
<b>5</b>	<b>ANEXOS .....</b>	<b>15</b>

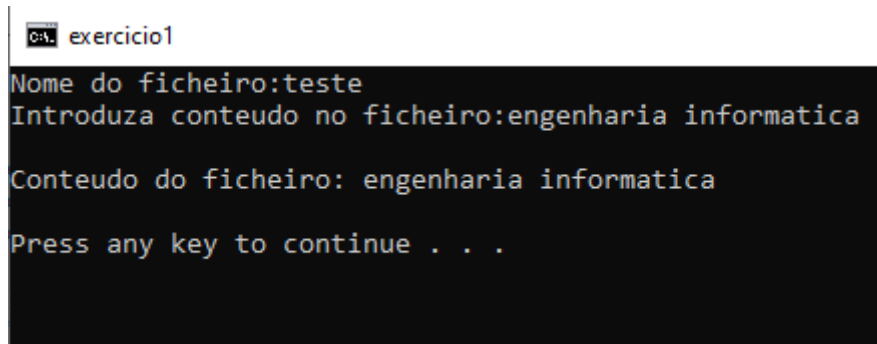
## 1 Introdução

Este trabalho foi feito no âmbito da disciplina de Estruturas de dados e tem como objetivo abordar conhecimentos sobre dois tipos de ficheiros, ficheiros texto e ficheiros binários.

## 2 Trabalho realizado

### 2.1 Pergunta 1

No primeiro exercício ,ao executarmos o programa, vai-nos perguntar que nome queremos dar ao ficheiro e em seguida pergunta-nos que conteúdo queremos introduzir no ficheiro ,e após isso apresenta-nos o conteúdo do ficheiro como podemos verificar na imagem abaixo:



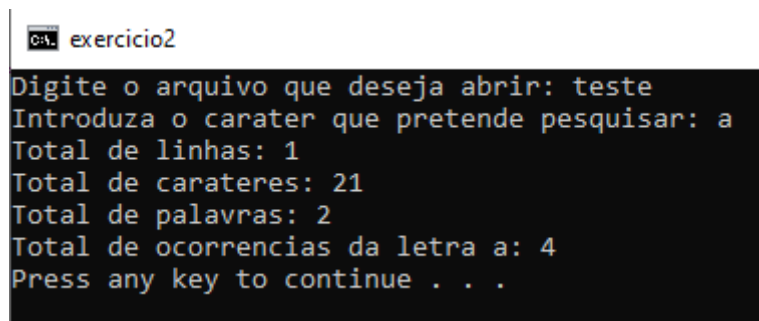
```
exercicio1
Nome do ficheiro:teste
Introduza conteudo no ficheiro:engenharia informatica
Conteudo do ficheiro: engenharia informatica
Press any key to continue . . .
```

*Figura 1-Exemplo da Pergunta 1*

## 2.2 Pergunta 2

O exercício 2 tinha como objetivo abrir um ficheiro a pedido do utilizador e indicar o número de linhas , o número de palavras, o número de caracteres e o número de ocorrências de um carácter indicado pelo utilizador.

Para isso, vai nos perguntar qual é o ficheiro que desejamos abrir e depois pergunta qual o carácter que pretendemos procurar ,posteriormente aparece-nos o número de linhas , o número de palavras, o número de caracteres e o número de ocorrências de um carácter indicado pelo utilizador, como tinha escrito anteriormente.



```
C++: exercicio2
Digite o arquivo que deseja abrir: teste
Introduza o carater que pretende pesquisar: a
Total de linhas: 1
Total de carateres: 21
Total de palavras: 2
Total de ocorrencias da letra a: 4
Press any key to continue . . .
```

*Figura 2-Exemplo da Pergunta 2*

## 2.3 Pergunta 3

Neste exercício 3 basicamente temos que alterar o exercício 2, mas no final do ficheiro temos que adicionar informação das horas em que foram registadas as alterações das alíneas.

Para isso utilizámos a biblioteca <time.h> e as funções time(...), ctime(...).

A função time tem como objetivo extrair a hora e a função ctime retorna a data e a hora

```
time(&hora);
```

Figura 3-Função time(...)

```
,ctime(&hora));
```

Figura 4-Função ctime(...)

```
1 engenharia informatica
2 Total de linhas: 1
3 Total de caracteres: 21
4 Total de palavras: 2
5 Total de ocorrencias da letra a: 4
6 Registo de alterações: Sat Apr 11 23:52:27 2020
```

Figura 5-Ficheiro teste

## 2.4 Pergunta 4

Neste exercício 4 pergunta-se ao utilizador quantos utilizadores quer ter no seu ficheiro (100 || 500||1000) a partir daí crias 3 ficheiros txt um com os números por ordem decrescente , outro com os números por ordem crescente e outro com os números ordenados aleatoriamente.

Neste exemplo usamos 100 números:

```
cresc100.txt
1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
```

Figura 5-Ficheiro ordem crescente

```
deces100.txt
1 100 99 98 97 96 95 94 93 92 91 90 89 88 87 86 85 84 83 82 81 80 79 78 77 76 75 74 73 72 71 70 69
```

Figura 6-Ficheiro ordem decrescente

```
aleat100.txt
1 42 68 35 1 70 25 79 59 63 65 6 46 82 28 62 92 96 43 28 37 92 5 3 54 93 83 22 17 19 96 48
```

Figura 7-Ficheiro ordenado aleatoriamente

## 2.5 Pergunta 5

No exercício 5 para o resolvermos temos que criar os ficheiros no exercício 4 com 100, 500 e 1000 números, a partir daí executamos o programa e automaticamente cria um ficheiro com os 50 números do ficheiro aleat100.txt, os 100 primeiros do aleat500.txt e os 500 primeiros números do aleat1000.txt.

Depois cria um ficheiro com os números ímpares do ficheiro inicialmente criado e cria um ficheiro com os números pares do ficheiro inicialmente criado.

primeirosnum.txt																																															
1	42	68	35	1	70	25	79	59	63	65	6	46	82	28	62	92	96	43	28	37	92	5	3	54	93	83	22	17	19	96																	
•	127	324	38	39	119	83	430	42	42	468	335	501	170	725	479	359	963	465	706	146	282	828	962																								
•	291	162	637	356	768	656	575	32	53	351	151	942	725	967	431	108	192	8	338	458	288	754	38																								
•	512	203	635	273	56	329	647	363	887	876	434	870	143	845	417	882	999	323	652	22	700	558																									
•	23	169	19	788	906	959	392	203	626	478	415	315	825	335	875	373	160	834	71	488	298	519	1																								

Figura 8-Ficheiro dos primeiros números

impares.txt																																																							
1	85	1	25	79	59	63	65	43	37	5	3	93	83	17	19	27	39	13	95	23	65	69	335	1	225	479	359	4																											
•	987	291	637	575	53	351	151	725	967	431	759	589	423	947	507	31	169	901	763	411	625																																		
•	169	19	959	203	415	315	825	335	875	373	71	519	271	669	193	103	481	803	925	973	433																																		

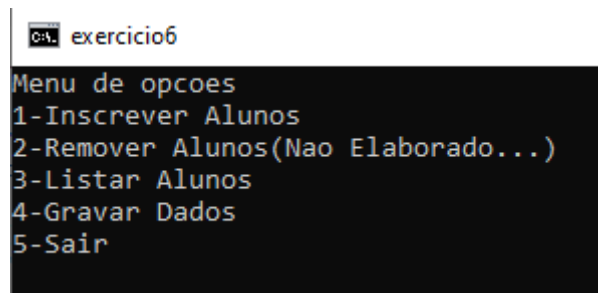
Figura 9-Ficheiro dos números ímpares, dos números do ficheiro inicial

		pares.txt																						
1		42	68	70	6	46	82	28	62	92	96	28	92	54	22	96	48	72	70	68	100	36	4	:
•		162	356	768	656	32	942	108	192	8	338	458	288	754	384	946	910	210	:					:
•		650	314	788	906	392	626	478	160	834	488	298	178	774	764	986	214	6	:					:

Figura 10-Ficheiro dos números pares, dos números do ficheiro inicial

## 2.6 Pergunta 6

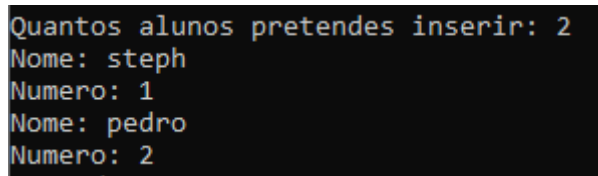
Neste exercício o objetivo era fazer um programa que permitisse inscrever alunos numa turma, que listasse os alunos inscritos e remover um aluno e também que a lista de alunos continuasse entre as várias execuções do programa. Basicamente é parecido com o trabalho final da cadeira de Fundamentos de Programação elaborado no 1º semestre. Para este programa usámos um menu de 5 opções:



```
exercicio6
Menu de opcoes
1-Inscrever Alunos
2-Remover Alunos(Nao Elaborado...)
3-Listar Alunos
4-Gravar Dados
5-Sair
```

Figura 11-Menu de opções

A opção 1 inscrever alunos, vai perguntar ao utilizador quantos alunos pretende inserir e após pergunta o nome e o número do aluno.

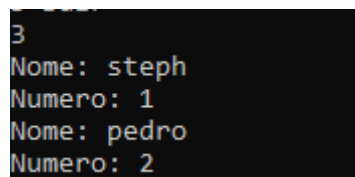


```
Quantos alunos pretendes inserir: 2
Nome: steph
Numero: 1
Nome: pedro
Numero: 2
```

Figura 12-Exemplo da opção inscrever

A opção 2 remover alunos, tivemos um pouco dificuldade a elaborar por isso achámos melhor não fazer.

A opção 3 listar alunos como o nome indica lista os alunos inscritos



```
3
Nome: steph
Numero: 1
Nome: pedro
Numero: 2
```

Figura 13-Exemplo da opção listar



Na opção 4 gravar dados, basicamente grava os dados dos alunos nome e número num ficheiro txt .

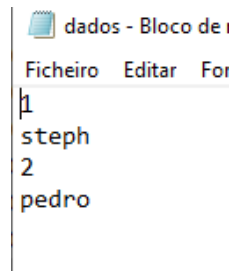


Figura 14-Ficheiro de dados

Na opção 5 sai do programa.

Ao voltar a executar o programa ,os dados do ficheiro criado na opção 4 são carregados.

## 2.7 Pergunta 7

Neste exercício ao executarmos vai nos pedir quantos nomes queremos introduzir e depois aparece a perguntar o nome para introduzir na tabela.

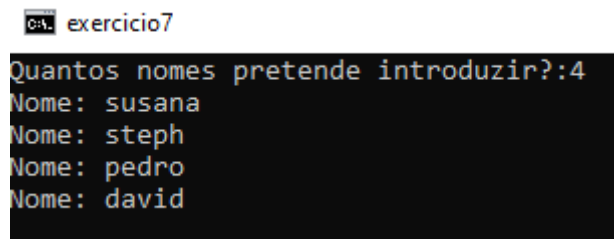


Figura 15-Exemplo de introdução de nomes

Apos isso são criados 3 ficheiros onde são apresentados no terminal :

-Um ficheiro .txt com a tabela anterior criada

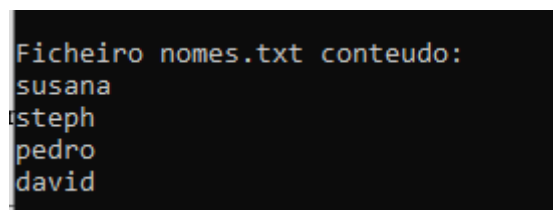


Figura 16-Conteudo do ficheiro nomes.txt

-Um ficheiro .dat que se cria da mesma forma que um ficheiro txt só que se acrescenta um b no modo de abertura do ficheiro.

```
FILE *f=fopen("tabelaNomes.dat","wb");
```

Figura 17-Modo de abertura de um ficheiro .dat

-Então cria um ficheiro .dat onde guarda a tabela de uma só vez

```
Ficheiro tabelaNomes.dat conteudo:  
susana  
steph  
pedro  
david
```

Figura 18-Conteúdo da tabelaNomes.dat

-E cria um ficheiro .dat onde guarda cada um dos nomes da tabela

```
Ficheiro nomesBinario.dat conteudo:  
susana  
steph  
pedro  
david
```

Figura 19-Conteúdo do nomesBinario.dat

## 2.8 Pergunta 8

No exercício 8 basicamente é o mesmo código que o exercício 6 só que em vez de guardar os dados num ficheiro de texto vai guardar num ficheiro binário .dat

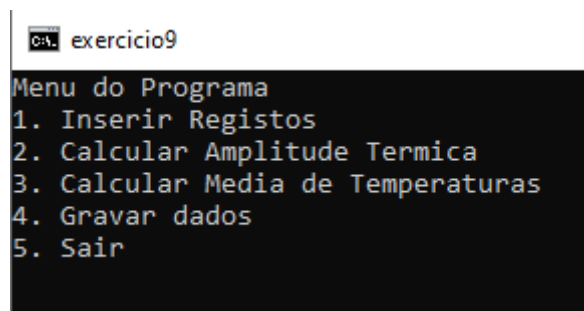
```
void guardar(char nome[][TAM], char numero[][TAM],int talunos){  
  
    FILE *dados=fopen("dados.dat","ab");  
  
    for (int i=0; i < talunos; i++){  
        fwrite(numero[i], sizeof(numero[i]),1, dados);  
        fwrite(nome[i], sizeof(nome[i]),1, dados);  
    }  
    fclose(dados);  
}
```

Figura 20-Função guardar dados em ficheiro binário

## 2.9 Pergunta 9

Neste exercício pede-nos para guardar num ficheiro binário os dados recolhidos por um termómetro digital automático que regista a temperatura de 15 em 15 minutos, onde os dados são inseridos pelo utilizador, também nos pede para calcular a amplitude térmica de um dia assim como a média das temperaturas registadas.

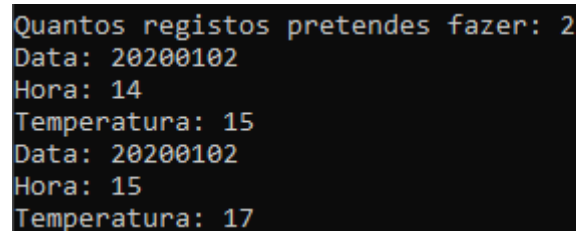
Para isso usámos um menu de 5 opções:



```
C:\> exercicio9
Menu do Programa
1. Inserir Registos
2. Calcular Amplitude Termica
3. Calcular Media de Temperaturas
4. Gravar dados
5. Sair
```

Figura 21-Menu de opções

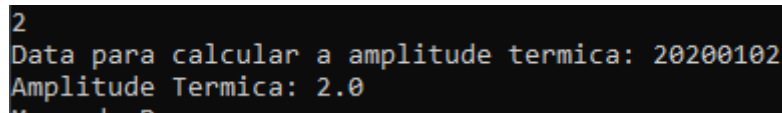
Na opção 1 inserir registos , vai perguntar ao utilizador quantos registos pretende fazer e em seguida pergunta a data, hora e temperatura do registo



```
Quantos registos pretendes fazer: 2
Data: 20200102
Hora: 14
Temperatura: 15
Data: 20200102
Hora: 15
Temperatura: 17
```

Figura 22-Exemplo da opção 1

Na opção 2 calcular a amplitude térmica, o programa vai perguntar qual é a data em que pretende calcular a amplitude térmica e em seguida apresenta a amplitude térmica



```
2
Data para calcular a amplitude termica: 20200102
Amplitude Termica: 2.0
```

Figura 23-Exemplo da opção 2

Na opção 3 calcular media de temperaturas, o programa vai perguntar qual é a data em que pretende calcular a media de temperaturas e em seguida apresenta a media de temperaturas

```
3
Data para calcular a media das temperaturas: 20200102
Media da temperatura: 16.0
```

Figura 24-Exemplo da opção 3

Na opção 4 o programa guardas os registos num ficheiro binário chamado de temperaturas.dat



Figura 25-Ficheiro temperatura.dat

Na opção 5 sai do programa

Ao voltar a executar o programa ,os registos do ficheiro criado na opção 4 são carregados.

### 3 Conclusão

Neste trabalho conseguimos corresponder as nossas expectativas, apenas tivemos um breve entrave em elaborar a função de remover aluno , mas no geral esta ficha de trabalho deu para revermos conceitos aprendidos anteriormente e começar a aprender sobre ficheiros binários. Foi um trabalho desafiador de se fazer.

## 4 Referências

<http://www.w3big.com/pt/cplusplus/c-function-ctime.html>

[https://help.highbond.com/helpdocs/analytics/141/scripting-guide/pt-br/Content/lang\\_ref/functions/r\\_time.htm](https://help.highbond.com/helpdocs/analytics/141/scripting-guide/pt-br/Content/lang_ref/functions/r_time.htm)

Ficha de trabalho nº2 disponibilizada pela professora Célia

## 5 Anexos

Pergunta 1– exercício1.c

Pergunta 2– exercício2.c

Pergunta 3– exercício3.c

Pergunta 4– exercício4.c

Pergunta 5– exercício5.c

Pergunta 6– exercício6.c

Pergunta 7– exercício7.c

Pergunta 8– exercício8.c

Pergunta 9– exercício9.c