

Relatório do trabalho prático de Programação Imperativa

Curso - Licenciatura em Engenharia de Sistemas Informáticos

Luís Esteves - 16960
João Morais - 17214
Sérgio Ribeiro - 18858

Barcelos - 20 de janeiro de 2022

Conteúdo

1	Introdução	2
2	Desenvolvimento	3
2.1	main	3
2.2	Func.c	4
2.3	Func2.c	4
2.4	header.h	6
3	Problemas encontrados	8
4	Conclusão	8

1 Introdução

Este trabalho prático foi realizado no âmbito da disciplina de Programação Imperativa, inserida no plano de estudos do curso de Engenharia de Sistemas Informáticos, onde se pretende desenvolver soluções na linguagem C.

Esta aplicação tem como objetivo desenvolver uma solução digital que permita agilizar a gestão de meios de mobilidade urbana num contexto de Smart-city, criando uma interface que possibilite ao utilizador fazer pedidos, para andar num dado transporte.

Este pedido tem de incluir o tempo que o utilizador usará, e a distância que irá percorrer. Por sua vez temos um ficheiro que irá guardar os meios de mobilidade disponíveis e finalmente irá ser preciso distribuir os meios de mobilidade consoante os pedidos feitos.

Para desenvolver esta aplicação utilizamos o editor de código Visual Studio Code.

2 Desenvolvimento

Para o desenvolvimento deste projeto começamos por criar:

- main;
- Func.c;
- Func2.c;
- header.h;

2.1 main

Aqui é o ponto de partida para a execução do nosso programa, aqui iremos fazer a leitura e inserção de dados nas structs e o acesso ao menu que terá todas as opções que o enunciado pede.

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <locale.h>
5  #include "Header.h"
6  #define TAMANHO 100 // Definição da constante TAMANHO com valor 100
7
8  //Main.c -> Main do nosso projecto;
9
10 int main (){
11
12     //Inicialização de algumas variáveis;
13     FILE *nomeFicheiro = fopen("ficheiro1.txt", "r");
14     struct Pedidos p1[TAMANHO]; // cria um vetor de 100 pedidos.
15     struct Transportes t1[TAMANHO]; // cria um vetor de 100 transportes.
16     struct Utilizacao u1[TAMANHO];
17     int i, f;
18     i = f = 0;
19     setlocale(LC_ALL, "Portuguese");
20
21     if (nomeFicheiro == NULL)
22     {
23         wprintf(L"Erro! Não foi encontrado nenhum ficheiro\n");
24         exit(-1);
25     }
26
27     //Leitura e inserção dos dados na struct;
28     while (!feof(nomeFicheiro))
29     {
30         fscanf(nomeFicheiro,"%d %d %s %f %f\n", &p1[i].ordem, &p1[i].NIF, p1[i].codigo, &p1[i].tempo, &p1[i].distancia);
31         i++;
32     }
33     fclose(nomeFicheiro);
34
35     FILE *nomeFicheiro2 = fopen("ficheiro2.txt", "r");
36     if (nomeFicheiro2 == NULL)
37     {
38         wprintf(L"Erro! Não foi encontrado nenhum ficheiro\n");
39         exit(-1);
40     }
41
42     //O mesmo processo mas agora para outro ficheiro;
43     while (!feof(nomeFicheiro2))

```

```

44 {
45     fscanf(nomeFicheiro2,"%s %s %f %f\n", t1[f].codigo, t1[f].tipo, &t1[f].preco, &t1[f].autonomia);
46     f++;
47 }
48 fclose(nomeFicheiro2);
49
50 //Menu que vai ter todas as opções que o projecto pede;
51 //Aqui vamos dar a opção de o utilizador inserir a opção que quer, depois, é claro, de ler os ficheiros de texto;
52 Menu(t1,p1,u1);
53
54     return 0;
55 }

```

2.2 Func.c

Este ficheiro conterá algumas funções que utilizaremos ao longo do projeto, em especial a função do menu que irá conter todas as funcionalidades que o projeto pede como, por exemplo:

- Inserção de dados;
- Remoção de dados;
- Listagem de dados;
- Armazenamento de dados;

```

----- Menu -----
3. Inserção de um novo meio de mobilidade elétrica;
4. Remoção de um meio de mobilidade elétrica a partir do seu código;
5. Inserção de um novo pedido de utilização;
6. Remoção de um pedido de utilização a partir do seu código;
7. Listagem dos dados de todos os meios de mobilidade elétrica;
8. Listagem de todos os pedidos de utilização;
9. Cálculo do custo associado a um pedido de utilização a partir da indicação do seu número de ordem;
10. Distribuição dos meios de mobilidade pelos vários utilizadores;
11. Listagem do plano de utilização de um determinado meio de mobilidade elétrica;
12. Armazenamento dos pedidos de utilização em ficheiro;
13. Armazenamento dos meios de mobilidade elétrica em ficheiro;
14. Sair;
Digite a opção que quer ver:

```

Figura 1: Menu

2.3 Func2.c

Este ficheiro foi criado com o propósito de guardar funções, para facilitar a compreensão e o funcionamento do projeto.

Temos neste ficheiro funções para ler individualmente os 3 ficheiros txt guardando os seus valores na struct e retornando o seu numero de linhas preenchidas, tendo tambem uma função para comparar os códigos que existem no Ficheiro2.txt e uma possível inserção no mesmo.

```

1 int Ler_Fich1(struct Pedidos p1[]){
2     FILE *nomeFicheiro2 = fopen("ficheiro1.txt", "r");
3     int i = 0;
4     if (nomeFicheiro2 == NULL)
5     {
6         wprintf(L"Erro! Não foi encontrado nenhum ficheiro\n");
7         exit(-1);
8     }
9
10    while (!feof(nomeFicheiro2))
11    {
12        fscanf(nomeFicheiro2,"%d %d %s %f %f\n", &p1[i].ordem, &p1[i].NIF, p1[i].codigo, &p1[i].tempo, &p1[i].distancia);

```

```

13     i++;
14 }
15 fclose(nomeFicheiro2);
16
17     return (i);
18 }
19
20 int Ler_Fich2(struct Transportes t1[]){
21     FILE *nomeFicheiro2 = fopen("ficheiro2.txt", "r");
22     int f = 0;
23     if (nomeFicheiro2 == NULL)
24     {
25         wprintf(L"Erro! Não foi encontrado nenhum ficheiro\n");
26         exit(-1);
27     }
28
29     while (!feof(nomeFicheiro2))
30     {
31         fscanf(nomeFicheiro2, "%s %s %f %f\n", t1[f].codigo, t1[f].tipo, &t1[f].preco, &t1[f].autonomia);
32         f++;
33     }
34     fclose(nomeFicheiro2);
35
36     return (f);
37 }
38
39 int Ler_Fich3(struct Utilizacao u1[]) {
40
41     FILE* nomeFicheiro2 = fopen("ficheiro3.txt", "r");
42     int f = 0;
43     setlocale(LC_ALL, "Portuguese");
44     if (nomeFicheiro2 == NULL)
45     {
46         wprintf(L"Erro! Não foi encontrado nenhum ficheiro\n");
47         exit(-1);
48     }
49
50     while (!feof(nomeFicheiro2))
51     {
52         fscanf(nomeFicheiro2, "%d %d %d %d %d %s\n", &u1[f].ordem, &u1[f].NIF, &u1[f].tempoInicio, &u1[f].tempoFim, &u1[f].au
53         f++;
54     }
55     fclose(nomeFicheiro2);
56
57     return (f);
58 }
59
60 int Comparar_Strings(char codigo [], int count, struct Transportes t1[], int tipo){
61     int i = 0;
62     int verificar = 1;
63     //No caso de tipo = 0, então a função vai retornar apenas se a palavra existe ou não no array;
64     if (tipo == 0){
65         for (i = 0; i <= count; i++){
66             if (verificar != 0) verificar = strcmp(codigo, t1[i].codigo);
67         }
68         return verificar;
69     }
70     else if (tipo == 1){
71         int aux = 0;

```

```

72     for (i = 0; i <= count; i++){
73         if (verificar != 0) {
74             verificar = strcmp(codigo, t1[i].codigo);
75             aux = i;
76             if(verificar == 0){
77                 memset(t1[aux].codigo,0,strlen(t1[aux].codigo));
78                 t1[aux].autonomia = 0;
79                 t1[aux].preco = 0;
80                 memset(t1[aux].tipo,0,strlen(t1[aux].tipo));
81             }
82         }
83     }
84 }
85 return aux;
86 }
87 }

```

2.4 header.h

Aqui é o Header do projeto, contem as structs que vamos utilizar no nosso projecto e as funções que utilizaremos ao longo do projeto.

Essas structs são a struct de Pedidos onde contem o numero da ordem do pedido, o NIF do utilizador, o código de mobilidade pretendida, tempo de utilização pretendido e a distância, temos também a struct Transportes onde contém o código, tipo, preço e a autonomia, sendo a struct Utilizacao uma junção das structs Pedidos e Transportes.

```

1  struct Pedidos
2  {
3      int ordem;
4      int NIF;
5      char codigo[5];
6      float tempo;
7      float distancia;
8  };
9
10 //Struct que vai guardar meios de transportes disponíveis;
11 struct Transportes
12 {
13     char codigo[5];
14     char tipo[20];
15     float preco;
16     float autonomia;
17 };
18
19 struct Utilizacao{
20     int ordem;
21     int NIF;
22     int tempoInicio;
23     int tempoFim;
24     int autonomiaInicial;
25     char codigo[5];
26 };
27
28
29 //Funções que vamos utilizar ao longo do projecto;
30 //Manipulação de strings;
31 char* strcpy(char* destination, const char* source);

```

```

32 int strcmp (const char* str1, const char* str2);
33 //Menu;
34 void Menu(struct Transportes t1[], struct Pedidos p1[], struct Utilizacao u1[]);
35 //Ponto 3
36 void Insercao_Dados_Fich2(struct Transportes t1[]);
37 //Ponto 4
38 void Eliminar_Fich2(struct Transportes t1[]);
39 //Ponto 5
40 void Insercao_Dados_Fich1(struct Pedidos p1[]);
41 //Ponto 6
42 void Eliminar_Fich1(struct Pedidos p1[]);
43 //Ponto 7
44 void Listar_Fich2(struct Transportes t1[]);
45 //Ponto 8
46 void Listar_Fich1(struct Pedidos p1[]);
47 //Ponto 9
48 void Calculo_Custo(struct Pedidos p1[], struct Transportes t1[]);
49 //Ponto 10
50 void Distribuir_Pedidos(struct Pedidos p1[], struct Transportes t1[], struct Utilizacao u1[]);
51 //Ponto 11
52 void Listar_Fich3(struct Utilizacao u1[]);
53 //Ponto 12
54 void Save_Fich1(struct Pedidos p1[]);
55 //Ponto 13
56 void Save_Fich2(struct Transportes t1[]);
57
58 //-----
59
60 //Funções que foram feitas para ler ficheiros de texto;
61 int Ler_Fich1(struct Pedidos p1[]);
62 int Ler_Fich2(struct Transportes t1[]);
63 int Ler_Fich3(struct Utilizacao u1[]);
64
65 //Faz alguma manipulação de strings;
66 int Comparar_Strings(char codigo [5], int count, struct Transportes t1[], int tipo);

```


3 Problemas encontrados

Durante a realização deste trabalho, deparámo-nos com dificuldades em atualizar os ficheiros corretamente e na distribuição dos meios de mobilidade pelos vários utilizadores.

4 Conclusão

Em jeito de conclusão,o trabalho realizado foi vantajoso para a unidade curricular, uma vez que permitiu por em prática a matéria na disciplina de uma maneira mais prática e num contexto real.

O trabalho permitiu-nos melhorar a nossa capacidade a nível de manipulação de ficheiros de texto, manipulação de strings, no uso de arrays, variaves e structs, permitindo desta forma melhorar a nossa experiência com a linguagem de programação utilizada.

O trabalho da unidade curricular não foi apenas vantajoso para consolidar matéria abordada, como também nos ensinou novos métodos que nos permitiu trabalhar em grupo com bastante sucesso e motivação.