

Métodos de Programação 2

MiETI :: 1º Ano

2018/2019



- Eduardo Paiva, 85312
eduardopaiva@live.com.pt



- Tiago Ferreira, 85392
tiago.ferreira.19@hotmail.com



Índice

Introdução.....	3
Enunciado.....	4
Discussão crítica da solução.....	5
Código em C.....	6
Exemplos de utilização da solução.....	17
Conclusão.....	18

INTRODUÇÃO

Este projeto realizado no âmbito da cadeira de Métodos de Programação II visa implementar um programa de gestão de uma Biblioteca (ou Arquivo) temática.

Neste relatório iremos apresentar a descrição do problema, bem como a nossa solução para o problema proposto e exemplos da sua implementação.

ENUNCIADO

Neste trabalho prático pretende-se que implementem um programa de gestão de uma Biblioteca (ou Arquivo) temática.

A temática da biblioteca e a caracterização de cada documento incluído no repositório documental associado à biblioteca fica ao critério de cada grupo.

A aplicação deve permitir as seguintes operações sobre os documentos do repositório:

- carregar (ler) os dados relativos a cada documento, os quais serão inseridos direta e manualmente no terminal pelo operador;
- remover um documento;
- visualizar os dados dum determinado documento ou então obter uma listagem completa de todo o repositório (provavelmente, sem mostrar todos os dados de cada documento);
- manter, ou seja, corrigir um ou mais dados dum documento.

Notar que devem usar uma lista ligada para guardar em memória todos os documentos arquivados. Depois de criar o seu repositório documental, a aplicação deve permitir:

- processar requisições de documentos, uma de cada vez, inseridas manualmente pelo operador no terminal. Cada requisição é uma lista de documentos requisitados, indicando-se para cada um o seu código e a quantidade pedida;
- verificar se a requisição é exequível, ou seja, se está disponível no arquivo a quantidade suficiente de cada item;
- actualizar o repositório, marcando como emprestada a quantidade solicitada e a pessoa/entidade que fez a requisição.

DESCRIÇÃO DA SOLUÇÃO

Em primeiro lugar, através do enunciado retiramos as informações mais importantes.

Verificamos logo que na nossa solução iríamos utilizar uma estrutura onde iríamos guardar os dados de cada livro, neste caso o ISBN, nome, quantidade disponível e dados das encomendas

Optamos por criar funções para responder as necessidades pedidas, uma para adicionar livros, uma para editar livros, uma para apagar livros e uma para consultar a biblioteca. Criamos também um pequeno menu que permite ao utilizador utilizar as funcionalidades do nosso programa.

No final acrescentamos uma função para guardar os dados introduzidos e outra para carregar os dados para serem utilizados numa secção posterior.

CÓDIGO

```
/* MP2 - G23 */
/* Eduardo Paiva A85312 */
/* Tiago Ferreira A85392 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX 50

FILE *dados;

typedef struct Biblioteca{

    int codigo;
    int ISBN;
    char nome[MAX];
    int quantidade;
    char pessoa[MAX];
    int pessoaEnc;
    struct Livro *prox;

}livro;

livro *ListaInicio=0;
livro *atual=0;
livro *fim=0;

livro *crialista(int IS,int cod, char nom[], int quan, char pess[], int pes){

    livro *lista = (livro*) malloc(sizeof(livro));

    lista->codigo=cod;
    lista->ISBN=IS;
```

```
        strcpy(lista->nome, nom);
        lista->quantidade=quan;
        lista->pessoaEnc=pes;
        lista->prox = 0;

        ListalInicio=atual=lista;

        return(lista);

    }

    livro *adiLivro(int IS, char nom[], int quan, char pess[], int pes){

        livro *lista = (livro*) malloc(sizeof(livro));
        lista->ISBN=IS;
        strcpy(lista->nome, nom);
        lista->quantidade=quan;
        lista->pessoaEnc=pes;
        lista->prox = 0;

        atual->prox=lista;
        atual=lista;

        return lista;

    }

    livro *procura(int IS){

        livro *procu=ListalInicio;
        int encon=0;

        while(procu!=0){

            if(procu->ISBN==IS){
                encon=1;
                break;
            }
        }
    }
```



```
        }else{
            fim=procu;
            procu=procu->prox;
        }
    }

    if(encon==1){

        return procu;
    }else{
        return 0;
    }

}

livro *editarLivro(livro**esco){

    livro*editar= *esco;
    int escol=0;
    char nom[MAX];
    int IS=0;
    int quan;

    printf("\nNome->1\nISBN->2\nQuantidade->3\n");
    scanf("%d", &escol);
    system("cls");

    if(escol==1){

        printf("Nome:");
        fflush(stdin);
        fgets(nom,MAX,stdin);
        strcpy(editar->nome,nom);

    }if(escol==2){
```

```
printf("ISBN:");
scanf("%d", &IS);
editar->ISBN=IS;

}if(escol==3){

printf("Quantidade:");
scanf("%d", &quan);
editar->quantidade=quan;

}

}

livro *apaga(livro**esco){

livro *atual=*esco;
if (fim!=0){
fim->prox=atual->prox;
}
if (atual==ListInicio){
ListInicio=atual->prox;
}
free(atual);
atual=0;
}

livro *requisitar(char pess[],int quan, livro**esco){

livro *requis= *esco;

if(requis->quantidade<quan){

printf("Nao existe unidades suficientes na biblioteca\n");
```

```
    }else{

        strcpy(requis->peessoa, pess);
        requis->peessoaEnc=quan;
        requis->quantidade=(requis->quantidade)-quan;
    }

}

void imprimir(){

    livro *aux=ListalInicio;

    while(aux!=0){

        printf("Nome:%sISBN:%d\nQuantidade:%d\n\n",    aux->nome,    aux->ISBN,    aux->quantidade);

        if(aux->peessoaEnc!=0){

            printf("Requisicao de %d livros por parte de %s\n", aux->peessoaEnc, aux->peessoa);

        }

        aux= aux->prox;
    }
}

void ficheiro(FILE *dados) {

    dados=fopen("biblioteca.txt","w");

    livro *gravar=ListalInicio;

    while (gravar!=0){

        fprintf(dados,"%d\n%s%d\n%d\n%s%d", gravar->codigo, gravar->nome, gravar->ISBN,
        gravar->quantidade, gravar->peessoa, gravar->peessoaEnc);

        gravar=gravar->prox;
    }
}
```

```
}
```

```
void carregarbibli(FILE *dados){

    int IS;
    int cod;
    int quan;
    char nom[MAX];
    char pess[MAX];
    int cont=0;
    int pes;
    int i=0;

    dados=fopen("biblioteca.txt","r");

    if (dados==0){
        return 0;
    }

    while (!feof(dados)){

        fscanf(dados,"%d\n",&cod);
        fgets(nom,MAX,dados);
        fscanf(dados,"%d\n",&IS);
        fscanf(dados,"%d\n",&quan);
        fgets(pess,MAX,dados);
        fscanf(dados,"%d\n",&pes);

        if (cont==0){

            crialista(IS,cod,nom,quan,pess, pes);
            cont++;
        }else{

            adiLivro(IS, nom,quan, pess, pes);
        }
    }
}
```

```
        cod++;
    }
}

int main()
{
    int IS;
    int cod=0;
    int quan;
    char nom[MAX];
    char pess[MAX];
    int cont=0;
    int pes=0;

    carregarbibli(dados);
    livro *lista=ListalInicio;
    livro *esco=0;

    int esc;
    while(esc != 0){

        system("cls");
        printf("\n -----\\n");
        printf("|  Adicionar Livro:    1 |\\n");
        printf("|  Editar Livro:      2 |\\n");
        printf("|  Apagar Livro:      3 |\\n");
        printf("|  Requisitar livro:   4 |\\n");
        printf("|  Consultar Biblioteca: 5 |\\n");
        printf("|  Sair e guardar dados: 0 |\\n");
        printf(" -----\\n");
        scanf("%d", &esc);

        switch(esc){

            case 1:
                system("cls");
```

```
if(cont==0 && lista==0){

    printf("Nome:");
    fflush(stdin);
    fgets(nom,MAX,stdin);
    printf("ISBN:");
    scanf("%d", &IS);
    printf("Quantidade:");
    scanf("%d", &quan);
    crialista(IS,cod, nom, quan, pess, pes);
    cod++;
    cont++;
    system ("cls");

}else{

    while(lista!=0){

        IS=lista->ISBN+1;
        lista=lista->prox;

    }

    printf("Nome:");
    fflush(stdin);
    fgets(pess,MAX,stdin);
    printf("ISBN:");
    scanf("%d", &IS);
    printf("Quandtidade:");
    scanf("%d", &quan);
    adiLivro( IS, nom, quan, pess, pes);
    cod++;
    system ("cls");
}

break;
```

case 2:

```
        system("cls");
        system("cls");
        printf("Escreva o ISBN do livro que quer editar:");
        scanf("%d",&IS);
        esco=procura(IS);
        if(esco==0){

            printf("\n Nao encontrado \n");

        }else{

            editarLivro(&esco);

        }
        int b=0;
        printf("\n\nVoltar(1)");
        scanf("%d", &b);
        if(b==1){

            break;

        }
        break;
case 3:

        system("cls");
        printf("Escreva o ISBN do livro que quer apagar:");
        scanf("%d",&IS);
        esco=procura(IS);

        if(esco==0){

            printf("\n Nao encontrado \n");

        }else{

            apaga(&esco);
```

```
        printf("Livro apagado");

    }

    printf("\n\nVoltar(1)");
    int c=0;
    scanf("%d", &c);
    if(c==1){

        break;

    }

    break;
case 4:

    system("cls");
    printf("Nome Pessoa:");
    fflush(stdin);
    fgets(pess,MAX,stdin);
    printf("ISBN:");
    scanf("%d", &IS);
    printf("Quantidade:");
    scanf("%d", &quan);

    esco=procura(IS);

    requisitar(pess,quan,&esco);
    int f=0;
    imprimir();
    printf("\n\nVoltar(1)");
    scanf("%d", &c);
    if(f==1){

        break;

    }
}
```



```
        break;
    case 5:

        system("cls");
        int e=0;
        imprimir();
        printf("\n\nVoltar(1)");
        scanf("%d", &c);
        if(e==1){

            break;

        }
        break;
    case 0:

        ficheiro(dados);

        break;
    }

}

}
```

EXEMPLOS DE UTILIZAÇÃO DA SOLUÇÃO

Menu inicial:

```
-----  
| Adicionar Livro:      1 |  
| Editar Livro:        2 |  
| Apagar Livro:        3 |  
| Requisitar livro:     4 |  
| Consultar Biblioteca: 5 |  
| Sair e guardar dados: 0 |  
-----
```

Adicionar livro:

```
Nome:Lord of the rings  
ISBN:11  
Quantidade:23
```

Apagar livro:

```
Escreva o ISBN do livro que quer apagar:11  
Livro apagado  
Voltar(1)_
```

Requisitar livro:

```
Nome Pessoa:tiago  
ISBN:12  
Quantidade:2  
Nome:Lord  
ISBN:12  
Quantidade:106  
Requisicao de 2 livros por parte de tiago  
Voltar(1)
```

CONCLUSÃO

Através da solução que desenvolvemos foi possível resolver o problema apresentado de forma eficaz e precisa.

O processo mais trabalhoso foi a passagem do algoritmo para código.

Uma das partes que tivemos mais dificuldades foi a realização das encomendas e carregamentos dos ficheiros.

Foi um projeto positivo, pois foi possível consolidar os conhecimentos de listas e apontadores que adquirimos ao longo do semestre.