

Arquivos Binários – Estrutura Física

Livros.dat



1	Linguagem C	38.50
2	Estrutura de Dados	89.30
3	Redes de Computadores	110.10
4	Implementando Redes Neurais	122.20



Registro



```
struct reg_livro{  
    int codigo;  
    char titulo[50];  
    float preco;  
};
```

Arquivos Binários – Estrutura Física

Livros.dat

1	Linguagem C	38.50
2	Estrutura de Dados	89.30
3	Redes de Computadores	110.10
4	Implementando Redes Neurais	122.20



livrosnew.dat

1	Linguagem C	38.50
2	Estrutura de Dados	89.30
4	Implementando Redes Neurais	122.20



Estrutura da Aplicação

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
```

Includes das Bibliotecas utilizadas

```
struct reg_livro{
    int codigo;
    char titulo[50];
    float preco;
};
```

Estrutura dos Registros do Arquivo

```
void cadastrarLivro() {
```

```
void listarLivros() {
```

Funções que implementam as Funcionalidades

```
int main() {
    int op;
```

```
do{
```

```
    printf("\n\n::::::::::::::::::::: Livraria Saber :::::::::::::::::::::::");
    printf("\n:: 1) Cadastrar Livro ::");
    printf("\n:: 2) Listar Todos os Livros ::");
    printf("\n:: 3) Consultar Livro peloCodigo ::");
    printf("\n:: 4) Consular Livro pelo Titulo ::");
    printf("\n:: 0) Sair ::");
    printf("\n::");
    printf("\n Opcao-> ");
    fflush(stdin); scanf("%d",&op);
```

Menu Principal
Menu Principal

```
switch(op) {
    case 1: //Cadastrar um novo Livro
        cadastrarLivro();
        break;
    case 2: //Relatorio de todos os Livros
        listarLivros();
        break;
}
```

Switch case que chama as funções da Aplicação

```
}while (op!=0);
```

```
//Fim main()
```



Cadastrar

Algoritmo

1. Entrada dos dados do Livro
2. Abrir o arquivo
3. Gravar os dados
4. Fechar o Arquivo

```
void cadastrarLivro() {  
    FILE *fplivros;  
    struct reg_livro livro;  
    char opc;  
  
    //Solicita os dados  
    printf("\nDigite o codigo do Livro: ");  
    fflush(stdin); scanf("%i",&livro.codigo);  
    printf("Digite o Titulo do Livro: ");  
    fflush(stdin); gets(livro.titulo);  
    printf("Digite o Preco do Livro: ");  
    fflush(stdin); scanf("%f",&livro.preco);  
  
    printf("\nGravar? (S/N) ");  
    fflush(stdin); scanf("%c",&opc);  
  
    if ((opc=='S') || (opc=='s')) {  
        //Abrir o Arquivo  
        fplivros = fopen("livros.dat","ab");  
        //Gravar  
        fwrite(&livro, sizeof(livro), 1, fplivros);  
        //Fechar o Arquivo  
        fclose(fplivros);  
        printf("\nLivro Cadastrado com Sucesso.");  
    }  
    else  
        printf("\nOperacao Cancelada!");  
} //Fim cadastrarLivro()
```



Entendendo o Código

Para transferir os dados digitados para o arquivo no disco, precisamos de uma variável ponteiro (*) tipo FILE

```
FILE *fplivros;  
struct reg_livro livro;
```

Os dados digitados pelo usuário serão armazenados temporariamente na memória na variável livro, do tipo struct reg_livro.

```
scanf("%i",&livro.codigo);  
gets(livro.titulo);  
scanf("%f",&livro.preco);
```

Cada informação digitada será armazenada no respectivo campo na variável livro, do tipo struct reg_livro.

Antes de transferir os dados para o Arquivo no disco, é necessário abrir o arquivo, dizendo o nome do arquivo (livros.dat) e o modo de abertura (ab). Neste caso, como foi especificado somente o nome do arquivo, ele será gravado no mesmo diretório do .exe do programa. O modo de abertura define qual a operação será realizada no arquivo. Neste caso, (ab) significa adicionar (a) dados em arquivo binário (b)

```
fplivros = fopen("livros.dat", "ab");
```

Grava os dados da memória (&livro) para o arquivo (fplivros).

```
fwrite(&livro, sizeof(livro), 1, fplivros);
```

```
fclose(fplivros);
```

Fecha o arquivo



Listar todos os Livros

```
void listarLivros() {  
    FILE *fplivros;  
    struct reg_livro livro;  
  
    //Abrir o Arquivo  
    fplivros = fopen("livros.dat", "rb");  
  
    //Ler cada registro e mostrar na tela  
    while (fread(&livro, sizeof(livro), 1, fplivros) == 1) {  
        printf("\n%3d %-30s %8.2f", livro.codigo, livro.titulo, livro.preco);  
    }  
  
    //Fechar o arquivo  
    fclose(fplivros);  
}  
//Fim listarLivros()
```

- Aqui é importante notar que o modo de abertura do arquivo ("rb") é para leitura do arquivo. Isso faz com que o ponteiro seja posicionado no primeiro registro.
- O laço de repetição vai ler registro por registro até atingir o fim do arquivo que, quando atingido, faz com que o fread retorne zero, pois não vai conseguir ler um registro, terminando o laço.



Consulta pelo Código do Livro

```
void consultarPeloCodigo() {
FILE *fplivros;
struct reg_livro livro;
int cod; //codigo a ser Localizado
int achou=0;

printf("\nDigite oCodigo do Livro a ser Localizado: ");
fflush(stdin); scanf("%i",&cod);

//Abrir o Arquivo
fplivros = fopen("livros.dat","rb");

//Ler cada registro e comparar com o codigo a ser localizado
while ((!achou)&&(fread(&livro,sizeof(livro),1,fplivros) ==1)) {
    if (livro.codigo==cod) {
        printf("\n%3d %-30s %8.2f", livro.codigo, livro.titulo,livro.preco);
        achou=1;
    }
}

if (achou==0) {
    printf("\nCodigo de Livro NAO existente.");
}

//Fechar o arquivo
fclose(fplivros);
} //Fim consultarPeloCodigo()
```



Entendendo o Código

Abrir o arquivo para Leitura ("rb"). Isso posiciona o ponteiro no primeiro registro.

```
//Abrir o Arquivo  
fplivros = fopen("livros.dat", "rb");
```

Sairá do laço de repetição se encontrar o livro com o código procurado (!achou) ou atingir o final do arquivo (fread())

```
while ((!achou) && (fread(&livro, sizeof(livro), 1, fplivros) == 1)) {  
    if (livro.codigo == cod) {  
        printf("\n%3d %-30s %8.2f", livro.codigo, livro.titulo, livro.preco);  
        achou = 1;  
    }  
}
```

Note que ao encontrar o Livro, a variável achou vai mudar para 1.



Consulta pelo Título do Livro

```
void consultarPeloTitulo() {
FILE *fplivros;
struct reg_livro livro;
char tit[50]; //Guarda o Titulo do Livro a ser Localizado
int achou=0;

printf("\nDigite o Titulo do Livro a ser Localizado: ");
fflush(stdin); gets(tit);

//Abrir o Arquivo
fplivros = fopen("livros.dat", "rb");

//Ler cada registro e comparar com o Titulo a ser localizado
//Obs.: A função strcmp() compara duas strings e retorna zero se forem iguais
while ((!achou) && (fread(&livro, sizeof(livro), 1, fplivros) == 1)) {
    if (strcmp(livro.titulo, tit) == 0) {
        printf("\n%3d %-30s %8.2f", livro.codigo, livro.titulo, livro.preco);
        achou=1;
    }
}
if (achou==0) {
    printf("\nLivro NAO existente.");
}
//Fechar o arquivo
fclose(fplivros);
} //Fim consultarPeloTitulo()
```

A diferença aqui é que o título é string e sua comparação é feita pela função strcmp()



Consulta por Palavra chave no Título do Livro

```
void consultarPalavraChaveTitulo(){
FILE *fplivros;
struct reg_livro livro;
char palavra[50]; //Guarda a palavra chave digitada pelo usuário
int achou=0;

printf("\nDigite a Palavra chave: ");
fflush(stdin); gets(palavra);

//Abrir o Arquivo
fplivros = fopen("livros.dat", "rb");

//Ler cada registro e verificar se o Título contém a palavra chave
//Obs.: A função strstr() retorna NULL se não encontrar a palavra dentro do Título
while (fread(&livro, sizeof(livro), 1, fplivros) == 1) {
    if (strstr(livro.titulo, palavra) != NULL) {
        printf("\n%3d %-30s %8.2f", livro.codigo, livro.titulo, livro.preco);
        achou=1;
    }
}
if (achou==0){
    printf("\nNenhum Livro possui a Palavra chave.");
}
//Fechar o arquivo
fclose(fplivros);
} //Fim consultarPalavraChaveTitulo()
```

A diferença aqui é que devemos percorrer todos os registros do arquivo, pois devemos imprimir todos os livros que contenham no Título a palavra chave procurada.



Consulta por Palavra no Início do Título do Livro

```
void consultarPalavraInicioTitulo(){
FILE *fplivros;
struct reg_livro livro;
char palavra[50]; //Guarda a palavra chave digitada pelo usuário
int achou=0, n;

printf("\nDigite a Palavra chave: ");
fflush(stdin); gets(palavra);

n = strlen(palavra); //retorna a qtde. de caracteres da string palavra

//Abrir o Arquivo
fplivros = fopen("livros.dat","rb");

//Ler cada registro e verificar se o Título contém a palavra chave no início
//Obs.: A função strncmp() retorna zero se os n primeiros caracteres forem iguais
while (fread(&livro, sizeof(livro), 1, fplivros) == 1) {
    if (strncmp(livro.titulo, palavra, n) == 0) {
        printf("\n%3d %-30s %8.2f", livro.codigo, livro.titulo, livro.preco);
        achou=1;
    }
}
if (achou==0) {
    printf("\nNenhum Livro Inicia com a Palavra chave.");
}
//Fechar o arquivo
fclose(fplivros);
} //Fim consultarPalavraInicioTitulo()
```

Mesma ideia do anterior, percorrer todos os registros do arquivo, porém, a função `strncmp()` compara somente os `n` primeiros caracteres das duas strings.



Atividade da Aula

- ✓ Melhorar a Consulta pelo início do Título de modo que seja independente de Maiúsculo e Minúsculo.



Dica: converta para maiúsculo ou minúsculo tanto a palavra digitada pelo usuário quanto o título do livro e depois faça a comparação.

clientes.dat

1	Joao Manuel Silva	1499878877	joao@gmail.com
2	Maria Oliveira Romaneli	1488019754	maria@htomail.com

vendas.dat

1	Nro. do Pedido
1	Código do Cliente
2	Código do Livro
1	Quantidade
2	
2	
3	
2	
3	
1	
4	
1	
3	
1	
1	
1	

Livros.dat

1	Linguagem C	38.50
2	Estrutura de Dados	89.30
3	Redes de Computadores	110.10
4	Implementando Redes Neurais	122.20