



UNIVERSIDADE FEDERAL DE MINAS GERAIS

Instituto de Ciências Exatas - ICEX

Departamento de Ciência da Computação – DCC

LISTA 3 – PDROGRAMAÇÃO E DESENVOLVIMENTO DE SOFTWARE 1

Aluno: Marcone Márcio da Silva Faria

Matrícula: 2019021573

Exercício 1

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

int main(){
    char caractere;
    int cont=0;
    FILE *arq;
    arq = fopen("file.txt","r");

    if(arq != NULL){
        while((caractere = fgetc(arq)) != EOF) {
            if (caractere == 'a' || caractere == 'A'){
                cont++;
            }
        }
        fclose(arq);
        printf("O arquivo tem %d caracteres do tipo 'a' ou 'A'", cont);
    }
}
```

Exercício 2

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

#define N 5

int main(){
    int mat1[N][N], mat2[N][N], soma[N][N];
    int result, i,j,k, cont=0;
    FILE *arq;

    for(i=0; i<N; i++){
        for(j=0; j<N; j++){
            mat1[i][j] = rand()%10;
            mat2[i][j] = rand()%10;
        }
    }

    printf("\nMatriz 1:\n");
```

```

for(i=0; i<N; i++){
    for(j=0; j<N; j++){
        printf("%d\t", mat1[i][j]);
    }
    printf("\n");
}

printf("\nMatriz 2:\n");
for(i=0; i<N; i++){
    for(j=0; j<N; j++){
        printf("%d\t", mat2[i][j]);
    }
    printf("\n");
}

arq = fopen("arquivo.bin","wb");

if(arq == NULL){
    printf("Erro ao abrir o arquivo!!!");
}

for (i=0; i<N; i++){
    for (j=0; j<N; j++){
        fwrite(&mat1[i][j], sizeof(int), 1, arq);
    }
}

for (i=0; i<N; i++){
    for (j=0; j<N; j++){
        fwrite(&mat2[i][j], sizeof(int), 1, arq);
    }
}
fclose(arq);

arq = fopen("arquivo.bin" , "rb" );

if (arq == NULL) {
    printf("Erro ao ler o arquivo");
}

for (i=0; i<N; i++){
    for (j=0; j<N; j++){
        soma[i][j] = mat1[i][j] + mat2[i][j];
    }
}

printf("\nSoma das matrizes:\n");

for (i=0; i<N; i++){
    for (j=0; j<N; j++){
        printf("%d\t", soma[i][j]);
    }
    printf("\n");
}

```

```
    fclose(arq);

    return 0;

}
```

Exercício 3

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

#define MAX 90

char nome[50];
char telefone[20];
unsigned int idade;
FILE *arq;

int main(){
    unsigned int menu;

    printf("BEM VINDO A SUA AGENDA DE CONTATOS EM C!!!\n\n");

    printf("Escolha a opcao desejada:\n");
    printf("[1] - Inserir Contatos, [2] - Alterar Contato, [3] - Excluir Contato, [4] - Exibir Contatos\n");
    scanf("%d", &menu);

    switch(menu){
        case 1:
            inserir();
            break;
        case 2:
            alterar();
            break;
        case 3:
            excluir();
            break;
        case 4:
            exibir();
            break;
        default:
            printf("Opção Invalida!!!\n");
    }
}

void inserir(){
    int n;

    printf("Quantos contatos voce deseja incluir?\n");
    scanf("%d", &n);

    do{
        printf("Digite o nome:\n");
```

```

fflush(stdin);
fgets(nome, 50, stdin);

printf("Digite o telefone:\n");
fgets(telefone, 20, stdin);

printf("Digite a idade:\n");
scanf("%d", &idade);

arq = fopen("agenda.bin", "a+b");

if(arq == NULL){
    printf("Erro ao abrir o arquivo!!!");
}

fwrite(&nome[50], sizeof(char), strlen(nome), arq);
fwrite(",", sizeof(char), 1, arq);
fwrite(&telefone[20], sizeof(char), strlen(telefone), arq);
fwrite(",", sizeof(char), 1, arq);
fwrite(&idade, sizeof(int), 1, arq);
fclose(arq);
n--;
}while(n>0);
}

```

```

void alterar(){
    char altera[50], copia[50], buf[MAX];
    int cont=0;
    long int posicao=0;

    printf("Digite o nome do contato a ser alterado:\n");
    fflush(stdin);
    fgets(altera, 50, stdin);

    printf("Digite a alteracao a ser realizada:\n");
    fflush(stdin);
    fgets(copia, 50, stdin);

    arq = fopen("agenda.bin", "r+b");
    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    fgets(buf, MAX, arq);
    while (!feof(arq)){
        if(strcmp(altera, strtok(buf, ",")) == 0){
            strcpy(altera, copia);
            fseek (arq, posicao, SEEK_SET);

            fwrite(&altera[50], sizeof(char), strlen(altera), arq);
            break;
        }
        posicao = ftell(arq);
    }
    printf("Alteracao realizada com sucesso!!!");
}

```

```

    fclose(arq);
}

void excluir(){
    char del[50], buf[MAX];
    long int posicao=0;

    printf("Digite o nome do contato a ser excluido:\n");
    fflush(stdin);
    fgets(del, 50, stdin);

    arq = fopen("agenda.txt","r+");
    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    fgets(buf,MAX,arq);
    while (!feof(arq)){
        if(strcmp(del,(strtok(buf,""))) == 0){
            strcpy(del, NULL);
            strcpy(telefone, NULL);
            idade = NULL;

            fseek (arq, posicao, SEEK_SET);

            fwrite(&del[50],sizeof(char),strlen(del),arq);
            fwrite(&telefone[20],sizeof(char),strlen(telefone),arq);
            fwrite(&idade,sizeof(int),1,arq);
            break;
        }
        posicao = ftell(arq);
    }
    printf("Alteracao realizada com sucesso!!!");
    fclose(arq);
}

void exibir(){
    char *name, *tel, buf[MAX];
    int old;

    fseek(arq, 0, SEEK_SET);
    arq = fopen("agenda.bin", "rb");

    fgets(buf,MAX,arq);
    while (!feof(arq)){
        name = atoi(strtok(buf,""));
        tel = strtok(NULL,"");
        old = (strtok(NULL,""));
        printf("Nome: %s\nTelefone: %s\nIdade: %d\n",name, tel, old);
        fgets(buf,MAX,arq);
    }
    fclose(arq);
}

```

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

typedef struct agenda{
    char nome[50];
    char telefone[20];
    unsigned int idade;
}CONTATO;

struct agenda cadastro;
FILE *arq;

int main(){
    unsigned int menu;

    printf("BEM VINDO A SUA AGENDA DE CONTATOS EM C!!!\n\n");

    printf("Escolha a opcao desejada:\n");
    printf("[1] - Inserir Contatos, [2] - Alterar Contato, [3] - Excluir Contato, [4] - Exibir Contatos\n");
    scanf("%d", &menu);

    switch(menu){
        case 1:
            inserir();
            break;
        case 2:
            alterar();
            break;
        case 3:
            excluir();
            break;
        case 4:
            exibir();
            break;
        default:
            printf("Opção Invalida!!!\n");
    }
}

void inserir(){
    int n;

    printf("Quantos contatos voce deseja incluir?\n");
    scanf("%d", &n);

    do{
        printf("Digite o nome:\n");
        fflush(stdin);
        fgets(cadastro.nome, 50, stdin);

        printf("Digite o telefone:\n");
        fflush(stdin);
```

```

fgets(cadastro.telefone, 20, stdin);

printf("Digite a idade:\n");
scanf("%d", &cadastro.idade);

arq = fopen("agenda.bin", "a+b");

if(arq == NULL){
    printf("Erro ao abrir o arquivo!!!");
}
fseek(arq, 0, SEEK_END);

fwrite (&cadastro, sizeof ( CONTATO ), 1, arq);
n--;
}while(n>0);

fclose(arq);
printf("Dados Inseridos com sucesso!!!");
}

void alterar(){
    char altera[50], copia[50];
    int posicao=0;

    printf("Digite o nome do contato a ser alterado:\n");
    fflush(stdin);
    fgets(altera, 50, stdin);

    printf("Digite a alteracao a ser realizada:\n");
    fflush(stdin);
    fgets(copia, 50, stdin);

    arq = fopen("agenda.bin", "r+b");
    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    while (fread(&cadastro, sizeof (CONTATO), 1, arq ) == 1 ) {
        if(strcmp(altera, cadastro.nome) == 0 ){
            strcpy(cadastro.nome, copia);

            fseek (arq, posicao, SEEK_SET);

            fwrite(&cadastro, sizeof(CONTATO), 1, arq);
            break;
        }
        posicao = ftell(arq);
    }
    fclose (arq);
    printf("Dados Alterados com sucesso!!!");
}

void excluir(){
    char exclui[50];
    int posicao=0, i;

```

```

printf("Digite o nome do contato a ser excluído:\n");
fflush(stdin);
fgets(exclui, 50, stdin);

arq = fopen("agenda.bin", "r+b");
if(arq == NULL){
    printf("Erro ao abrir o arquivo!!!");
}

while (fread(&cadastro, sizeof (CONTATO), 1, arq ) == 1 ) {
    if(strcmp(exclui, cadastro.nome) == 0 ){
        strset(cadastro.nome);
        strset(cadastro.telefone);
        cadastro.idade = 0;

        fseek (arq, posicao, SEEK_SET);

        fwrite(&cadastro, sizeof(CONTATO), 1, arq);
        break;
    }
    posicao = ftell(arq);
}
fclose (arq);
printf("Dados Excluídos com sucesso!!!");
}

void exibir(struct agenda cadastro){
    arq = fopen("agenda.bin", "rb");

    while((fread(&cadastro, sizeof (CONTATO), 1, arq)) == 1 ) {
        printf ( "Nome: %s\n", cadastro.nome);
        printf ( "Telefone: %s\n", cadastro.telefone);
        printf ( "Idade: %d\n", cadastro.idade);
        printf ( "-----\n\n" );
    }
    fclose (arq);
}

```

Exercício 5

```

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

#define MAX 2

typedef struct dados{
    char nome[40];
    unsigned int num;
}t_dados;

FILE *arq;

t_dados vetor[MAX];

```



```

int main(){
    int i, menu=0;

    printf("Escolha a opcao desejada:\n");
    printf("[1] - Inserir Registro, [2] - Salvar Registro, [3] - Carregar Registro, [4] - Listar Registro, [5] -
Remover Registro\n");
    scanf("%d", &menu);

    switch(menu){
        case 1:
            inserir();
            break;
        case 2:
            salvar();
            break;
        case 3:
            carregar();
            break;
        case 4:
            listar();
            break;
        case 5:
            remover();
            break;
        default:
            printf("Opção Invalida!!!\n");
    }
}

void inserir(){
    int i,n;

    do{
        printf("Quantos registros voce deseja inserir?\n");
        scanf("%d", &n);
    }while(n>MAX);

    for(i=0; i<n; i++){
        printf("Digite o nome do %do registro:\n", i+1);
        fflush(stdin);
        fgets(vetor[i].nome, 40, stdin);

        printf("Digite o numero do %do registro:\n", i+1);
        scanf("%d", &vetor[i].num);

        printf("Registro inserido com sucesso na posicao %d do vetor de cadastros!!!\n", i);
    }
    salvar();
}

void salvar(){
    int i;
    arq = fopen("file.bin","a+b");

```

```

if(arq == NULL){
    printf("Erro ao criar o arquivo!!!");
}

for(i = 0; i < MAX; i++){
    fwrite(&vetor[i], sizeof(t_dados), 1, arq);
}
fclose(arq);
printf("\n\nDados gravados com sucesso!!!");
}

void carregar(){
    arq = fopen("file.bin", "r+b");

    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    for(k=0 ; k<MAX; k++){
        while((fread(&vetor[k], sizeof (t_dados), 1, arq)) == 1 ) {

        }
    }
}

void listar(){
    int k;

    arq = fopen("file.bin", "r+b");

    for(k=0 ; k<MAX; k++){
        while((fread(&vetor[k], sizeof (t_dados), 1, arq)) == 1 ) {
            if(vetor[k].num != 0){
                printf ( "Nome: %s", vetor[k].nome);
                printf ( "Idade: %d\n", vetor[k].num);
                printf ( "-----\n\n" );
            }
        }
    }
    fclose (arq);
}

void remover(){
    unsigned int ex;
    int i, posicao = 0;

    printf("Digite numero do registro a ser excluido:\n");
    scanf("%d", &ex);

    arq = fopen("file.bin", "r+b");
    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    for(i=0; i<MAX; i++){

```

```

while (fread(&vetor[i], sizeof (t_dados), 1, arq ) == 1 ) {
    if(ex == (vetor[i].num)){
        strset(vetor[i].nome);
        vetor[i].num = 0;

        fseek (arq, posicao, SEEK_SET);

        fwrite(&vetor[i],sizeof(t_dados),1,arq);
        break;
    }
    posicao = ftell(arq);
}
}
fclose (arq);
printf("Dados Excluidos com sucesso!!!");
}

```

Exercício 6

```

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

typedef struct lista{
    int info;
    struct lista *prox;
}Lista;

Lista *raiz = NULL;
int quebra;

Lista* constroi(int n, int *v);
Lista* separa (Lista* l, int n);

int main(){
    int n,*v,i, quebra;
    Lista *pnodo;

    printf("Digite a quantidade de elementos da lista encadeada:\n");
    scanf("%d", &n);

    v = (int *) malloc(n * sizeof(int));
    for(i=0; i<n; i++){
        v[i] = rand()%10;
    }

    raiz = constroi(n, v);

    printf("\n\nDigite um elemento da lista para separa-la:\n");
    scanf("%d", &quebra);

    separa(raiz, quebra);
}

Lista* constroi(int n, int *v){
    int i;
    Lista *inicio;

```

```

inicio = (Lista *)malloc(sizeof(Lista));

if(n == 0){
    return raiz;
}

for(i=0; i<n; i++){
    inicio->info = v[i];
    printf("%d\t", inicio->info);
    inicio->prox = raiz;
}
return inicio;
}

Lista* separa (Lista* l, int n){
    Lista *aux = l;

    while(aux->prox != NULL) {
        aux = aux->prox;
        printf("%d\t", aux->info);

        if(aux->info == n){
            break;
        }
    }
}

```

Exercício 7

```

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

typedef struct lista{
    int info;
    struct lista *prox;
}Lista;

Lista *raiz = NULL;

Lista* constroi(int n, int *v);

int main(){
    int n,*v,i;
    Lista *pnodo;

    printf("Digite a quantidade de elementos do vetor:\n");
    scanf("%d", &n);

    v = (int *) malloc(n * sizeof(int));
    for(i=0; i<n; i++){
        printf("Digite o %do elemento do vetor:\n", i+1);
        scanf("%d", &v[i]);
    }

    raiz = constroi(n, v);
    pnodo = raiz;

```

```

while(pnodo != NULL){
    printf("%d\n", pnodo->info);
    pnodo = pnodo->prox;
}
}

Lista* constroi(int n, int *v){
    int i;
    Lista *inicio;

    inicio = (Lista *)malloc(sizeof(Lista));

    if(n == 0){
        return raiz;
    }

    for(i=0; i<n; i++){
        inicio->info = v[i];
        printf("%d\n", inicio->info);
        inicio->prox = raiz;
    }
    return inicio;
}

```

Exercício 8

```

#include <stdio.h>
#include <stdlib.h>
typedef struct agenda{
    char nome[50];
    char telefone[20];
    struct agenda *prox;
} Agenda;

Agenda *contato;
FILE *arq;

int main(){
    unsigned int menu;

    printf("BEM VINDO A SUA AGENDA DE CONTATOS EM C!!!\n\n");

    printf("Escolha a opcao desejada:\n");
    printf("[1] - Inserir Contatos, [2] - Excluir Contato, [3] - Buscar Contato\n");
    scanf("%d", &menu);

    switch(menu){
        case 1:
            inserir();
            break;
        case 2:
            excluir();
            break;
        case 3:
            buscar(contato);

```

```

        break;
    default:
        printf("Opção Invalida!!!\n");
    }
}

void inserir(){
    int n;

    contato = (Agenda *) malloc(sizeof(Agenda));

    printf("Quantos contatos voce deseja incluir?\n");
    scanf("%d", &n);

    do{
        printf("Digite o nome:\n");
        fflush(stdin);
        fgets(contato->nome, 50, stdin);

        printf("Digite o telefone:\n");
        fflush(stdin);
        fgets(contato->telefone, 20, stdin);

        contato->prox = contato;

        arq = fopen("agenda.bin", "a+b");

        if(arq == NULL){
            printf("Erro ao abrir o arquivo!!!");
        }
        fseek(arq, 0, SEEK_END);

        fwrite (&contato, sizeof ( Agenda ), 1, arq);
        n--;
    }while(n>0);

    fclose(arq);
    printf("Dados Inseridos com sucesso!!!");
}

```

```

void excluir(){
    char exclui[50];
    int posicao=0, i;

    printf("Digite o nome do contato a ser excluido:\n");
    fflush(stdin);
    fgets(exclui, 50, stdin);

    arq = fopen("agenda.bin", "r+b");
    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    while (fread(&contato, sizeof (Agenda), 1, arq ) == 1 ) {

```

```

    if(strcmp(exclui, contato->nome) == 0 ){
        strset(contato->nome);
        strset(contato->telefone);
        contato->prox = contato;

        fseek (arq, posicao, SEEK_SET);

        fwrite(&contato,sizeof(Agenda),1,arq);
        break;
    }
    posicao = ftell(arq);
}
fclose (arq);
printf("Dados Excluidos com sucesso!!!");
}

void buscar(Agenda *cadastro){
    char busca[50];
    int i, posicao = 0;

    printf("Digite o contato a ser buscado\n");
    fflush(stdin);
    fgets(busca, 50, stdin);

    arq = fopen("file.bin", "r+b");
    if(arq == NULL){
        printf("Erro ao abrir o arquivo!!!");
    }

    while (fread(&cadastro, sizeof (Agenda), 1, arq ) == 1 ) {
        if(strcmp(busca, cadastro->nome) == 0){
            printf ( "Nome: %s", cadastro->nome);
            printf ( "Telfone: %s\n", cadastro->telefone);
            printf ( "-----\n\n" );
            fseek (arq, posicao, SEEK_SET);

        }
        posicao = ftell(arq);
    }
    fclose (arq);
}

```