



Disciplina:	Linguagens de Programação
Ano letivo:	2021-02
Carga horária:	60 horas
Professor:	Francisco de Assis S. Santos, Dr.
Acadêmicos:	

Atividade em sala: Análise de Código em C

Assumindo o código anexo, realize as seguintes análises:

- 1) Circule e defina os conceitos formais das Linguagens de Programação, com ênfase em expressões e comandos (tipos de operadores: Aridade e origem; tipos de expressões: literais, agregações, aritméticas, relacionais e booleanos)
- 2) No código em C entregue proponha a alteração da função de Pesquisa para a utilização de uma expressão condicional (Dica: pode ser usado o operador ternário)
- 4) Altere as estruturas agregativas extrato e correntista para a LP Java.

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

```
struct extrato{
    char data[30];
    char hora[30];
    char dia[3];
    char mes[4];
    char ano[5];
    float valor;
    char tipolancamento[40];
};
struct correntista{
    int conta, agencia, diaabertura, mesabertura, anoabertura, poslancamento;
    char nome[35];
    long CPF;
    float saldo;
    struct extrato lancamento[1000];
    int numAtualLancamentos;
};
struct correntista cliente[100];
```

```

int ultimaPosicao=0;

int entradaPesquisa(){
    int cpf;
    printf("\n Digite o CPF: ");
    scanf("%d",&cpf);

    return cpf; }

void cadastro(int posicao){

    __fpurge(stdin);
    printf("\n Digite o nome do cliente: ");
    scanf("%[^\\n]",cliente[posicao].nome);
    printf("\n Digite a conta: ");
    scanf("%d",&cliente[posicao].conta);
    printf("\n Digite a agencia: ");
    scanf("%d",&cliente[posicao].agencia);
    printf("\n Digite o CPF: ");
    scanf("%ld",&cliente[posicao].CPF);
    printf("\n Digite o dia da abertura da conta: ");
    scanf("%d",&cliente[posicao].diaabertura);
    printf("\n Digite o mes da abertura: ");
    scanf("%d",&cliente[posicao].mesabertura);
    printf("\n Digite o ano da abertura: ");
    scanf("%d",&cliente[posicao].anoabertura);
    printf("\n Digite o saldo inicial: ");
    scanf("%f",&cliente[posicao].saldo);
}

void imprimir(){
    int i=0;
    printf("\n\n Os dados informados dos clientes sao: \n\n");
    do{
        printf("\n NOME: %s \n", cliente[i].nome);
        printf("\n Conta: %d",cliente[i].conta);
        printf("\n Agencia: %d",cliente[i].agencia);
        printf("\n CPF: %ld \n",cliente[i].CPF);
        printf("\n DIA DA ABERTURA: %d",cliente[i].diaabertura);
        printf("\n MES ABERTURA: %d", cliente[i].mesabertura);
        printf("\n ANO DE ABERTURA: %d",cliente[i].anoabertura);
        i++;
    }while(i<ultimaPosicao);
}

int Pesquisa(int cpf){
    int i=0,achouPosicao=-1;
    while((achouPosicao==-1)&&(i<ultimaPosicao)){
        if(cliente[i].CPF==cpf){
            achouPosicao=i;
        }
        else
            i++;
    }
}

```

```

    return achouPosicao;
}

void mostrar(int pos){
    printf("\n NOME: %s \n", cliente[pos].nome);
    printf("\n Conta: %d", cliente[pos].conta);
    printf("\n Agencia: %d", cliente[pos].agencia);
    printf("\n DIA DA ABERTURA: %d", cliente[pos].diaabertura);
    printf("\n MES ABERTURA: %d", cliente[pos].mesabertura);
    printf("\n ANO DE ABERTURA: %d", cliente[pos].anoabertura);
}

float deposito(float valor, int pos){
    int poslancamento=cliente[pos].numAtualLancamentos;

    cliente[pos].saldo=cliente[pos].saldo+valor;
    strcpy(cliente[pos].lancamento[poslancamento].tipolancamento,"Deposito");

    strcpy(cliente[pos].lancamento[poslancamento].data,__DATE__);
    strcpy(cliente[pos].lancamento[poslancamento].hora,__TIME__);
    cliente[pos].lancamento[poslancamento].valor=valor;
    cliente[pos].numAtualLancamentos ++;

    return cliente[pos].saldo;
}

float saque(float valor, int pos){
    int poslancamento=cliente[pos].numAtualLancamentos;

    cliente[pos].saldo=cliente[pos].saldo-valor;
    strcpy(cliente[pos].lancamento[poslancamento].tipolancamento,"Saque");

    strcpy(cliente[pos].lancamento[poslancamento].data,__DATE__);
    strcpy(cliente[pos].lancamento[poslancamento].hora,__TIME__);
    cliente[pos].lancamento[poslancamento].valor=valor;
    cliente[pos].numAtualLancamentos ++;
    return cliente[pos].saldo;
}

void transferencia(int posOrigem, int posDestino){
    float valor;
    printf("\n Digite o valor a ser tranferido: ");
    scanf("%f",&valor);
    cliente[posOrigem].saldo=cliente[posOrigem].saldo-valor;
    cliente[posDestino].saldo=cliente[posDestino].saldo+valor;

    cliente[posOrigem].numAtualLancamentos ++;
    cliente[posDestino].numAtualLancamentos ++;
}

void extrato(int pos){
    int i;
    for(i=0;i<cliente[pos].numAtualLancamentos;i++){

```

```

        printf("\n Extrato: \n\n => Data: %s          hora: %s
\n\n",cliente[pos].lancamento[i].data, cliente[pos].lancamento[i].hora);
        printf("\ntipo de lancamento: %s          valor: %f",
cliente[pos].lancamento[i].tipolancamento,cliente[pos].lancamento[i].valor);
    }
}

```

```

int main(){

    int opcao,i,num,pos,pos2;
    float valor, valorSaque;
do{
    printf("\n Escolha uma opcao: ");
    printf("\n 1-Cadastrar clientes;");
    printf("\n 2-Imprimir clientes;");
    printf("\n 3-Pesquisar clientes;");
    printf("\n 4-Alterar cliente;");
    printf("\n 5-Imprimir saldo;");
    printf("\n 6-Deposito;");
    printf("\n 7-Saque;");
    printf("\n 8-Transferencia;");
    printf("\n 9-Imprimir extrato;");
    printf("\n 10-Sair;");
    scanf ("%d",&opcao);

    switch(opcao){
    case 1:
        cadastro(ultimaPosicao);
        ultimaPosicao++;
        break;
    case 2:
        imprimir();
        break;
    case 3:
        pos= Pesquisa(entradaPesquisa());
        if(pos!=-1)
            mostrar(pos);
        else
            printf("\n Elemento nao encontrado!");
        break;
    case 4:
        pos=Pesquisa(entradaPesquisa());
        if(pos!=-1){
            cadastro(pos);
        }
        else
            printf("\nElemento nao encontrado!");
        break;
    case 5:
        pos=Pesquisa(entradaPesquisa());
        if(pos!=-1){
            printf("\n Saldo: %f.",cliente[pos].saldo);

```

```

    }
    else
        printf("\n Elemento nao encontrado!");
    break;
case 6:
    pos=Pesquisa(entradaPesquisa());
    if(pos!=-1){
        printf("\n Digite o valor do deposito: ");
        scanf("%f",&valor);
        deposito(valor, pos);
    }
    else
        printf("\n Elemento nao encontrado!");

    break;
case 7:
    pos=Pesquisa(entradaPesquisa());
    if(pos!=-1){
        printf("\n Digite o valor do saque: ");
        scanf("%f",&valorSaque);
        saque(valorSaque, pos);
    }
    else
        printf("\n Elemento nao encontrado!");
    break;
case 8:
    pos=Pesquisa(entradaPesquisa());
    if(pos!=-1){
        pos2=Pesquisa(entradaPesquisa());
        if(pos2!=-1){
            transferencia(pos,pos2);
        }
        else
            printf("\n Elemento nao encontrado!");
    }
    else
        printf("\n Elemento nao encontrado!");
    break;
case 9:
    pos=Pesquisa(entradaPesquisa());
    if(pos!=-1){
        extrato(pos);
    }
    else
        printf("\n Elemento nao encontrado!");
    break;
case 10:
    exit(0);
default:
    printf("\n Valor invalido!\n");
}
}while(opcao!=10);
return 0;
}

```