**Atividades do conteúdo: Abertura**

**1-1** Desenvolva um programa C que, baseado no código a seguir, realize as seguintes tarefas:

int a, \*b;

1. Leia do usuário um valor a ser atribuído à variável **a**;
2. Faça o ponteiro **b** apontar para **a**;
3. Dobre o valor de **a**, utilizando o ponteiro **b**.
4. Aloque nova memória para **b**;
5. Leia do usuário um valor a ser atribuído à posição de memória apontada por **b**.
6. Mostre os valores de **a** e **b** na tela.

|  |
| --- |
| int a, \*b;    printf("Qual o valor atribuido a a?\n");  scanf("%d", &a);    b = (int\*)malloc(sizeof(int));  b = &a;  \*b = \*b \* 2;  printf("Valor de a: %d\nValor de b: %d\n", a, \*b);  b = (int\*)malloc(sizeof(int));  printf("Qual o valor atribuido a b?\n");  scanf("%d", &b);  b = &b;  printf("Valor de a: %d\nValor de b: %d, %d\n", a, \*b); |

**1 - 2** Implemente um TAD que represente uma carteira digital. Ele deverá conter o valor na carteira, além das operações de depósito e retirada e visualização do saldo.

|  |
| --- |
| carteira.h  typedef struct Carteira TCarteira;  TCarteira \* criar(float v);  void depositar(TCarteira \*c, float v);  void retirar(TCarteira \*c, float v);  float mostrar(TCarteira \*c);  carteira.c  #include <stdio.h>  #include <stdlib.h>  #include <stddef.h>  #include "carteira.h"  struct Carteira{    float valor;    };  TCarteira \* criar(float v){    TCarteira \*c = (TCarteira \*)malloc(sizeof(TCarteira));  c->valor = v;    return c;    }  void depositar(TCarteira \*c, float v){    c->valor = c->valor + v;    }  void retirar(TCarteira \*c, float v){    c->valor = c->valor - v;    }  float mostrar(TCarteira \*c){    printf("%.2f", c->valor);    }  main.c  int main(int argc, char \*argv[]) {    TCarteira \*cart;    cart = criar(0.0);    mostrar(cart);  printf("\n-----------------\n");  depositar(cart, 10.50);  mostrar(cart);  retirar(cart, 5.33);  printf("\n-----------------\n");  mostrar(cart);    return 0;  } |