**ESCOLA SUPERIOR DE TECNOLOGIA E EDUCAÇÃO DE RIO CLARO**

**ASSOCIAÇÃO DAS ESCOLAS REUNIDAS**

**Programação de Computadores II**

**Prof. Esp. Cristiano José Cecanho**

Lista de Exercícios - Implementação em ANSI C/C++ **obrigatória**

1. Procure determinar quais valores são impressos ao final deste programa. Confira sua resposta testando o programa no micro. Execute-o passo a passo conferindo o valor das variáveis em cada momento.
   1. void main()

{

int a,b,\*c;

a = 3;

b = 4;

c = &a; // c é igual ao endereço de memoria de a, ou seja não é o valor de a e sim o endereço da var a

b++;

\*c = a+2; /\* o ponteiro c é igual a somatoria de a + 2, 3 + 2

como lá em cima a gente declarou que c é igual ao endereço de memoria de a

ao colocarmos \*c = a + 2, nos mudamos o valor da variavel a\*/

printf("%d %d",a,b);

}

**RESPOSTA: 5 5**

* 1. void main()

{

int a,b,\*c;

a = 4;

b = 3;

c = &a; // c = 4

\*c = \*c +1; // 4 + 1 = 5 agora a vale 5

c = &b; // c = 3

b = b+4; // 3 + 4 = 7 = b é referencia de c, logo c vale 7 tambem

printf("%d %d %d",a,b,\*c); // 5 7 7

}

**RESPOSTA: 5 7 7**

* 1. void main()

{

a = 4;

b = 3;

c = &a; // c é igual ao endereço de memória de a

d = &b; // d é igual ao endereço de memória de b

\*c /= 2; // c = c/2 ... c = 2

f = c; //f = 2

c = d; // c = 3

d = f; // d = 2

printf("%d %d",\*c,\*d); // 3 2

}

**RESPOSTA: 3 2**

d. int calcula(int);

void main()

{

int a,b,c;

char d;

a=1;b=2;c=3;d='A';

a+=b\*c;

d=(a>7)?d-1:d+1;

b = calcula(b);

c = calcula(calcula(a));

a = c++;

printf("%d - %d - %d - %c\n",a,b,c,d);

}

int calcula(int x)

{

int i;

if ((x=x\*2)>5) return(x+3);

for(i=0;i<10;i++)

{

if (i<5) continue;

if (x>8) break;

x+=2;

}

return(x);

}

**38 10 B 37**

**7. Fazer um programa em "C" que lê 10 valores e imprime o maior e o menor valores lidos.**

#include <stdio.h>

#include <stdlib.h>

#define MAX 10

//7. Fazer um programa em "C" que lê 10 valores e imprime o maior e o menor valores lidos.

int main() {

int num;

int x=0, maior = -999, menor = 10000;

for(x=0 ; x<10 ; x++){

printf("Digite um numero: ");

scanf("%d", &num);

if(num > maior) maior = num;

if(num < menor) menor = num;

}

printf("Digito: %d %d ", maior , menor);

return 0;

}

**8. Fazer um programa que lê um conjunto de 10 valores e os imprime ordenados.**

*temp recebe o valor da cabeça da lista, enquanto o temp no campo prox for diferente de NULL é impresso na tela o conteúdo de temp no campo dado, e logo após, para correr o vetor temp recebe a próxima posição(temp=temp->prox). Isso vai se repetir até o temp->prox for igual a NULL, ou seja é o ultimo dado a ser impresso. Caso o flag no início da função for diferente de 1 aparecerá a mensagem: “Lista Vazia”*

#include <stdio.h>

#include <stdlib.h>

/\* 8. Fazer um programa que lê um conjunto de 10 valores e os imprime ordenados. \*/

void ordena(int \*num){ //colocando em ordem crescente

int x, j, temp;

for (x=0;x<9; x++){

for(j=x+1;j<10;j++){

if (num[x]>num[j]){ // se a posicao anterior for maior que a proxima

temp=num[x]; // posicao anterior fica em temp

num[x]=num[j]; // posicao anterior é substituida pela prox

num[j]=temp; // posicao proxima recebe o que estava em temp, ou seja invertemos as posicoes

}

}

}

}

int main() {

int num[10];

int x;

for(x=0; x<10; x++){

printf("Digite um numero: ");

scanf("%d", &num[x]);

}

ordena(num);

printf("ORDEM CRESCENTE:\n ");

for(x=0; x<10 ; x++){

printf("Posicao %d: %d\n ", x, num[x] );

}

return 0;

}

**9. Fazer uma rotina que recebe como parâmetro um array de 5 posições contendo as notas de um aluno ao longo do ano e devolve a média do aluno.**

#include <iostream>

#include <stdio.h>

#include<stdlib.h>

/\*9. Fazer uma rotina que recebe como parâmetro um array de 5 posições

contendo as notas de um aluno ao longo do ano e devolve a média do aluno. \*/

float calcula\_media(float \*notas){

float media;

media = (notas[1] + notas[2] + notas[3] + notas[4] + notas[5]) / 5;

return media;

}

int main() {

float notas[5];

float nota\_media;

int x;

for(x = 1; x<6; x++){

printf("Qual a %dª nota? ", x);

scanf("%f", &notas[x]);

}

nota\_media = calcula\_media(notas);

printf("\nA média é: %.2f\n", nota\_media);

return 0;

}

10. Fazer uma rotina que recebe um array do tipo double e o número de valores que devem ser solicitados ao usuário e devolve o array preenchido com os valores digitados.

11. Fazer um programa em "C" que lê um conjunto de 10 valores inteiros e verifica se algum dos valores é igual a média dos mesmos.

12. Fazer um programa que lê valores para uma matriz do tipo "float"de 5 linhas por 3 colunas e imprime a diferença entre a média dos elementos das colunas pares e a média dos elementos das linhas ímpares.

13. Fazer um programa em "C" que lê uma string qualquer de no máximo 80 caracteres e imprime:

- Quantos caracteres tem o string;

- Quantos caracteres são de pontuação;

- Quantos caracteres são números;

- Quantos caracteres são minúsculas.

14. Fazer uma função que retorna a soma, a diferença e o produto entre dois números.

15. Fazer uma função em "C" que retorna a razão entre dois números. A função deve retornar pelo comando return o valor 1 se a operação foi possível e o valor 0 se a operação não foi possível (divisão por zero, por exemplo). O resultado da divisão deve retornar por um parâmetro por referência.

16. Fazer uma rotina em "C" que recebe um vetor de números inteiros como parâmetro onde todos os valores exceto o último são positivos e devolve:

- a média dos valores do vetor;

- o menor valor do vetor (sem considerar o último)

- o maior valor do vetor.

17. Fazer uma função para ler e retornar o valor das 3 notas de um aluno.

18. Construir um programa em "C" que implementa uma agenda eletrônica. O programa deve ter um menu com as seguintes opções:

• Entrar um novo nome na agenda.

• Imprimir na tela os dados de uma das pessoas cadastradas (conforme

solicitação).

• Imprimir a lista de nomes cadastrados que comecem pela letra indicada.

• Fim

Cada entrada da agenda deve ter os seguintes campos:

char nome[30];

char endereco[100];

char fone[10];

long int CEP;

Obs: a agenda deve ter capacidade para 100 entradas.

19. Fazer um programa em "C" que lê uma lista de 20 produtos e preços armazená-los em um array do tipo da estrutura abaixo. O programa deve, em seguida, ordenar o vetor em ordem alfabética de nome de produto e inflacionar os produtos cujo valor for menor que 100 em 5%. Por fim a lista de produtos/preços deve ser impressa.

OBS: usar uma rotina que recebe uma estrutura do tipo PROD com parâmetro e atualiza o preço, uma que lê os dados para a estrutura do tipo PROD e outra capaz de imprimir a estrutura.

typedef struct

{

char nome[80];

float preco;

} PROD;

20. Construir um programa em "C" que implementa uma agenda eletrônica. O programa deve ter um menu com as seguintes opções:

1- Entrar um nome na agenda

2- Imprimir na tela os dados de uma das pessoas cadastradas (consulta por

nome)

3- Imprimir na impressora a lista dos nomes que começam pela letra indicada

https://www.ime.usp.br/~macmulti/exercicios/inteiros/index.html