**Nome:** Rafael Bazolli dos Santos

**Curso:** Análise e Desenvolvimento de Sistemas

**Disciplina:** Lógica de Programação

**Semestre / Ano:** 2º ADS / 2023

**Exercício - 01**

**Criar um programa que altera o valor pré-definido pelo programa através da**

**locação de memória.**

**Código:**

#include <stdio.h>

main()

{

int Y, \*P, X;

Y=0;

P=&Y;

X=\*P;

X=4;

(\*P)++;

X--;

(\*P)+=X;

printf("Y = %d \n", Y);

}

OBS: Acho válido ressaltar, a variável X, na linha 8, é inútil naquele ponto. É atribuído à ela um valor na linha 8, e na linha 9 é atribuído outro valor constante (no caso, o valor 4). Então a atribuição feita na linha 8 não teve efeito nenhum no código, ela poderia ser descartada.

**Exercício - 02**

**Criar um programa com ponteiro, que atribua duas constantes a duas variáveis do tipo inteiro e depois seja inserido o endereço das variáveis em dois ponteiros, imprimir os ponteiros, atribuir um ponteiro em outro e depois imprimir, depois, atribuir um número a um ponteiro já usado, também uma variável a outra e depois imprimir tudo.**

**Código:**

# include <stdio.h>

int main () {

int \* x, \* y;

int a, b;

a = 27;

b = 43;

x = &a;

y = &b;

printf("Valor: %d - %d \n\n", \*x , \*y ) ;

\*x=\*y;

printf("Valor: %d - %d \n\n", \*x , \*y ) ;

\*x = 27;

y = x;

printf ("Valor: %d - %d \n\n", \*x, \*y);

}

**Exercício - 03**

**Criar um programa que atribua os de valores em variáveis para exibição na tela.**

**Código:**

#include <stdio.h>

Int main()

{

// DECLARANDO AS VARIÁVEIS

int valor, \*p1, idade, vetor[3], \*p4, \*p5;

float temp, \*p2;

char aux, \*p3, \*nome = "Algoritmos";

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL VALOR PARA IMPRESSÃO NA TELA

valor = 15;

p1 = &valor;

\*p1 = 200;

printf("(a) %d \n", valor);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL TEMP PARA IMPRESSÃO NA TELA

temp = 26.5;

p2 = &temp;

\*p2 = 29.0;

printf("(b) %.1f \n", temp);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL AUX PARA IMPRESSÃO NA TELA

p3 = &nome[0];

aux = \*p3;

printf("(c) %c \n", aux);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL AUX PARA IMPRESSÃO NA TELA

p3 = &nome[4];

aux = \*p3;

printf("(d) %c \n", aux);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL P3 PARA IMPRESSÃO NA TELA

p3 = nome;

printf("(e) %c \n", \*p3);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL P3 PARA IMPRESSÃO NA TELA

p3 = p3 + 4;

printf("(f) %c \n", \*p3);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL P3 PARA IMPRESSÃO NA TELA

p3--;

printf("(g) %c \n", \*p3);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL IDADE PARA IMPRESSÃO NA TELA

vetor[0] = 20;

vetor[1] = 21;

vetor[2] = 22;

p4 = vetor;

idade = \*p4;

printf("(h) %d \n", idade);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL IDADE PARA IMPRESSÃO NA TELA

p5 = p4 + 1;

idade = \*p5;

printf("(i) %d \n", idade);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL IDADE PARA IMPRESSÃO NA TELA

p4 = p5 + 1;

idade = \*p4;

printf("(j) %d \n", idade);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL IDADE PARA IMPRESSÃO NA TELA

p4 = p4 - 2;

idade = \*p4;

printf("(l) %d \n", idade);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL P5 PARA IMPRESSÃO NA TELA

p5 = &vetor[2] - 1;

printf("(m) %d \n", \*p5);

//ATRIBUIÇÃO DE VALORES PARA DA VARIÁVEL P5 PARA IMPRESSÃO NA TELA

p5++;

printf("(n) %d \n", \*p5);

}

**Exercício - 04**

**Criar um programa que defina e exibe valores e endereços dos ponteiros na tela.**

**Código:**

#include <stdio.h>

#include <stdlib.h>

main()

{

float VET[5]={1.1,2.2,3.3,4.4,5.5}, \*F;

int I;

F = VET;

//ESPECIFICAÇÃO DO QUE SE TRATA OS VALORES IMPRESSOS PELO PROGRAMA

printf("Contador/valor/valor/endereco/endereco\n");

//COMANDO PARA REPETIR O VETOR CINCO VEZES

For (I=0; I<=4; I++)

{

printf("\nI = %d",I);

//PRIMEIRO VALOR

printf("\nVET[%d] = %.1f", I, VET[I]);

//SEGUNDO VALOR

printf("\n\*(F + %d) = %.1f",I, \*(F+I));

//PRIMEIRO ENDEREÇO

printf("\n&VET[%d] = %.X", I, &VET[I]);

//SEGUNDO ENDEREÇO

printf("\n(F + %d) = %.X",I, F+I);

printf("\n\n");

}

}

**Exercício - 05**

**Responda as perguntas abaixo, escolhendo a alternativa adequada para cada**

**questão.**

**1- Seja um vetor declarado por int vet[10]; Qual elemento deste vetor é acessado quando se escreve vet[2] ?**

a. Primeiro elemento

b. Segundo elemento

c. Terceiro elemento

d. Quarto elemento

e. Nenhuma das opções anteriores

**2- Se declararmos um vetor como: int vet[30], a instrução abaixo acessa corretamente os elementos deste vetor?**

**for (j=0; j <= 30; j++)**

**vet[j] = j\*j;**

a. Sim

b. Não

**3- Seja a matriz matrx declarada e inicializada por:**

**int matrx[][4] ={1,2,3,4,5,6,7,8,9,10,11,12};**

**O que conterá o elemento matrx[1][2] ?**

a. 2

b. 5

c. 6

d. 7

e. Nenhuma das opções anteriores

**4- Se uma string for declarada como: char str[20]; o número máximo de caracteres que poderão ser lidos e armazenados nela é:**

a. 18

b. 19

c. 20

d. 21

**5- Qual função pode ser usada para determinar o comprimento de uma string?**

a. gets

b. strcpy

c. strcat

d. strlen

e. strcmp

**6- Qual das instruções abaixo é correta para declarar um ponteiro para inteiro?**

a. \*int pti;

b. \*pti;

c. &i;

d. int\_pti pti;

e. int \*pti;

**7- Seja a seguinte sequência de instruções em um**

**programa C:**

**int \*pti;**

**int i = 10;**

**pti = &i;**

**Qual afirmativa é falsa?**

a. pti armazena o endereço de i

b. \*pti é igual a 10

c. ao se executar \*pti = 20; i passará a ter o valor 20

d. ao se alterar o valor de i, \*pti será modificado

e. pti é igual a 10

**8- Se i e j são variáveis inteiras e pi e pj são ponteiros para inteiro, qual atribuição é ilegal?**

a. pi = &i;

b. \*pj = &j;

c. pj = &\*&j;

d. i = \*&\*&j;

e. i = (\*pi)+++\*pj;

**9- Seja a seguinte seqüência de instruções em um programa C:**

**int \*pti;**

**int veti[]={10,7,2,6,3};**

**pti = veti;**

**Qual afirmativa é falsa?**

a. \*pti é igual a 10

b. \*(pti+2) é igual a 2

c. pti[4] é igual a 3

d. pti[1] é igual a 10

e. \*(veti+3) é igual a 6

**10- Na seqüência de instruções abaixo:**

**float f;**

**float \*pf;**

**pf = &f;**

**scanf("%f", pf);**

**O que está sendo feito?**

a. Efetuamos a leitura de f

b. Não efetuamos a leitura de f

c. Temos um erro de sintaxe

d. Deveríamos estar usando &pf no scanf

e. Nenhuma das opções anteriores

**11- Seja a seguinte seqüência de instruções:**

**Int i=10, j=20;**

**int \*pti, \*ptj;**

**pti = &i;**

**ptj = &j;**

**Qual expressão não é válida?**

a. j = pti == ptj;

b. i = pti-ptj;

c. pti += ptj;

d. pti++;

e. i = pti || ptj;

**12- Seja a declaração:**

**int matr[][4] = {1,2,3,4,5,6,7,8,9,10,11,12}**

**Qual afirmativa é falsa?**

a. \*\*matr é igual a 1

b. \*(\*(matr+1)+2) é igual a 7

c. \*(matr[2]+3) é igual a 12

d. (\*(matr+2))[2] é igual a 11

e. \*((\*matr)+1) é igual a 5