

Lista de Exercícios 09 – Linguagem C

CURSO: ENGENHARIA DA COMPUTAÇÃO – UFC – CAMPUS DE SOBRAL

DISCIPLINA: PROGRAMAÇÃO COMPUTACIONAL

PROFESSOR: FERNANDO RODRIGUES DE ALMEIDA JÚNIOR

ALUNO: _____ DATA: ____/____/2023

1. Indique se o código-fonte (programa) abaixo irá compilar corretamente ou não. Se sim, mostre o que será impresso após a execução do mesmo.

```
4 main(){
5     int n[] = {10,20}, *p;
6     p = n;
7
8     ++*p;
9     printf("%d %d\n",n[0],n[1]);
10    ++*p++;
11    printf("%d %d\n",n[0],n[1]);
12    ++*p--;
13    printf("%d %d\n",n[0],n[1]);
14    (++*p++)++;
15    printf("%d %d\n",n[0],n[1]);
16    (++*p--)++;
17    printf("%d %d\n",n[0],n[1]);
18    system("pause");
19 }
```

2. Ao final da seguinte sequência de instruções escritas em linguagem C:

```
int *pti;
int i = 10;
pti = &i;
```

Indique o valor de:

(*pti)--

*&i

++(*&*pti)

3. Na expressão

float *pont;

o que é do tipo float?

- a) A variável pont.
- b) O endereço de pont.
- c) A variável apontada por pont.
- d) A variável que aponta para pont.
- e) N.d.A.

4. Dadas as declarações abaixo, qual é o valor de cada um dos itens, considerando cada item independentemente dos anteriores, ou seja, de forma não cumulativa:

```
int x = 10, *px = &x, **ppx = &px;
float y = 5.9, *py = &y, **ppy = &py;
```

x	y	px	py	ppy	ppx
FFA0	FFB4	FFF0	FFC6	FFA6	FFD4

- | | | |
|----------|------------|--------------|
| a) x = | g) *ppx = | m) &ppy = |
| b) *py = | h) py = | n) *&px = |
| c) px = | i) &x = | o) **ppx++ = |
| d) &y = | j) py++ = | p) px++ = |
| e) *px = | k) *px-- = | q) &ppx = |
| f) y = | l) **ppy = | |

Dados os seguintes exercícios, escreva programas em C para resolver tais questões:

5. Crie um programa que declare um array de inteiros contendo cinco elementos. Utilizando apenas aritmética de ponteiros, leia esse array do teclado e imprima o dobro de cada valor lido.
6. Desenvolva uma função chamada **potencia** que receba dois números inteiros (x e y) e, sem utilizar outra função qualquer (nem a biblioteca *math.h*) e usando o princípio das multiplicações sucessivas para cálculo de potenciação, retorne o valor de x elevado a y. Escreva um programa que solicite ao usuário os respectivos valores dos parâmetros e chame a função criada para mostrar o resultado do cálculo ao usuário.
7. O fatorial de um número inteiro positivo N, denotado por N!, é definido como o produto dos inteiros positivos menores do que ou iguais a N. Por exemplo $4! = 4 \times 3 \times 2 \times 1 = 24$. Desenvolva uma função chamada **fatorial** que receba um número inteiro (x) e, sem utilizar outra função qualquer (nem a biblioteca *math.h*) retorne um inteiro longo sem sinal (*unsigned long long int*) constando do resultado do cálculo do fatorial do valor x. Escreva um programa que solicite ao usuário os respectivos valores dos parâmetros e chame a função criada para mostrar o resultado do cálculo ao usuário.
8. A seguinte sequência de números 0 1 1 2 3 5 8 13 21... é conhecida como *Série de Fibonacci*. Nessa sequência, cada número, depois dos 2 primeiros (fixados em 0 e 1), é igual à soma dos 2 anteriores. Escreva um algoritmo que defina uma função “*NFibonacci*” que receba como parâmetro um inteiro positivo N ($0 < N < 46$) e retorne um inteiro sendo o N-ésimo termo da *Série de Fibonacci*. No corpo da função “*main*”, leia um valor inteiro positivo (no mesmo padrão de N acima) e crie um laço que chame a função criada (*NFibonacci*) para os valores de 1 a N, mostrando os N primeiros números dessa série.
9. Dado que o fatorial de um número inteiro positivo N (ou seja, N!) é $N \times (N-1)!$, se N maior do que 1, e é igual a 1 se N = 0 ou N = 1, crie uma função que calcule o fatorial de um número recursivamente.
10. Dado que o valor de um N-ésimo elemento da *Série de Fibonacci* (fibo(N)) é igual a soma dos dois elementos anteriores ($fibo(N-1) + fibo(N-2)$) e que $fibo(1) = 0$ e $fibo(2) = 1$, implemente uma função recursiva que calcule o valor do N-ésimo termo da *Série de Fibonacci* e utilize esta função em um programa que leia um valor X inteiro positivo e exiba a série até aquele termo.

Bom trabalho!