

Disciplina	Curso	Turno	Período
Algoritmos e Estruturas de Dados I	Ciência da Computação	Manhã	1º
Professores:			Rosilane Mota e João Pedro Santos

## Lista 05 - Ponteiros

**Instruções:** Resolva os exercícios a seguir e entregue pelo **Canvas** um arquivo em formato **pdf**. Para as questões de programação, tire um print do código e cole no arquivo.

1. Explique cada uma das expressões a seguir, indicando a diferença entre elas:

p++;  
 (\*p)++;  
 \*(p++);

Qual informação se refere a expressão \*(p+10)?

2. Se o endereço de uma variável valor foi atribuído a um ponteiro valorPtr, quais alternativas são verdadeiras? Justifique sua resposta.
- valor == &valorPtr
  - valor == \*valorPtr
  - valorPtr == &valor
  - valorPtr == \*valor
3. Faça um programa que leia 3 números inteiros (a, b, c). Para cada valor lido, mostre o nome da variável, o endereço e o seu valor.
4. Implemente um programa em C que leia dois inteiros, armazenando-os em variáveis. O programa deve comparar os endereços das variáveis e exibir o maior deles.
5. Faça um algoritmo que declare uma variável e mostre o valor, endereço e tamanho na memória para cada um dos seguintes tipos: char, int, float, double, int \*, char \*, int \*, float \*, double \*. Dica: use a função sizeof(variável) para obter o tamanho da variável.
6. Implemente um programa em C que declare variáveis para armazenar um valor inteiro, um valor real e um caractere. Deve existir no programa ponteiros associados a cada um deles. O programa deve solicitar novos dados para as variáveis e elas devem ser modificadas usando os respectivos ponteiros. Exiba os endereços e os conteúdos de todas as variáveis e ponteiros, antes e após a alteração.
7. Implemente um programa que leia três valores inteiros e chame uma função que receba e ordene estes valores, ou seja, o menor valor na primeira variável, o segundo menor valor na variável do meio, e o maior valor na última variável. A função deve retornar o valor 1 se os três valores forem iguais e 0 se existirem valores diferentes. Exibir os valores ordenados na tela no programa principal.

8. Escreva uma função chamada divisores que:

- Recebe como parâmetro um número inteiro n por valor e dois números inteiros max e min por referência;
- Retorna 0 se n é primo e 1 caso contrário;
- Se n não for primo, as variáveis max e min devem assumir os valores do maior e do menor divisores inteiros de n, respectivamente. Desconsidere o número 1 e o próprio número n.

9. Identifique o erro no programa a seguir, de modo que seja exibido o valor 10 na tela.

```
1 #include <stdio.h>
2 int main()
3 {
4     int x, *p, **q;
5     p = &x;
6     q = &p;
7     x = 10;
8     printf("\n%d\n", &q);
9     return(0);
10 }
```

10. Observe os dois programas a seguir, Código 1 e Código 2. Qual é a diferença entre eles? Qual é o valor impresso para ptr em cada um dos códigos? Por quê?

```
1 //CÓDIGO 1
2 int main()
3 {
4     int *ptr, i;
5     ptr = (int *) malloc(sizeof(int));
6     *ptr = 10;
7     for(i=0;i<5;i++){
8         *ptr=*ptr+1;
9     }
10    printf("\nptr: %d", *ptr);
11    free(ptr);
12    return 0;
13 }

1 //CÓDIGO 2
2 int main()
3 {
4     int *ptr, i;
5     ptr = (int *) malloc(sizeof(int));
6     *ptr = 10;
7     for(i=0;i<5;i++){
8         ptr=ptr+1;
9     }
10    printf("\nptr: %p", ptr);
11    free(ptr);
12    return 0;
13 }
```