



## Lista de Exercícios 1

1. Descreva as principais características da Linguagem C.

A linguagem C foi criada em 1972, e se caracteriza por ser imperativa, procedural, compilada, multiplataforma e de alto nível.

2. Qual o significado do termo identificador na linguagem de programação C?

São termos criados pelo usuário para identificar “coisas”, podendo ser números ou letras, tanto maiúsculas quanto minúsculas.

3. Explique o que são tipos de dados. Apresente pelo menos três exemplos de como são utilizados.

A linguagem C possui alguns tipos de dados como caracteres, inteiros e pontos flutuantes.

Exemplos: char letra = B; int dia = 5; float divisao = 1,5;

4. A que se refere o termo pré-definidos na linguagem de programação C?

São termos que fazem parte do vocabulário da linguagem, e são para o uso da mesma.

5. Quais os três tipos de pré-definidos? Cite ao menos dois exemplos para cada tipo.

Palavras reservadas: “case” e “break”, Operadores: “==” e “&&”, Caracteres especiais: “{” e “,”.

6. Apresente o esqueleto básico que um programa em C precisa ter para funcionar (apresentado em sala de aula).

a. Demonstre, no programa criado, como as variáveis são declaradas;

b. Demonstre, no programa criado, uma operação envolvendo as variáveis declaradas.

```
Int main ()  
{  
    Int numero1, numero2, Soma;  
    numero1 = 12;  
    numero2 = 23;  
    Soma = numero1 + numero2;  
    return 0;  
}
```

7. Explique, com suas palavras, o que são bibliotecas na Linguagem C.

Bibliotecas são um conjunto de funções, que podem ser adicionadas a critério do usuário.

8. Explique o funcionamento dos comandos *printf* e *scanf*. Dê ao menos três exemplos de cada.

Scanf é usado para receber dados, por exemplo: `scanf("%d", &num1); scanf("%c", &letra3); scanf("%f", &raiz);`

Printf é usado para exibir dados, por exemplo: `printf("Hello World"); printf("\n %d", num2); printf("\n\n\t Numero Par");`

9. Apresente um programa contendo:

- O esqueleto básico para seu funcionamento;
- Declaração de variáveis e operação com as variáveis declaradas;
- Bibliotecas em C (úteis para o programa).
- printf* e *scanf*.

```
#include <stdio.h>
```

```
int main ()
```

```
{
```

```
int n1, n2;
```

```
printf("\n Digite o primeiro numero: ");
```

```
scanf("%d", &n1);
```

```
printf("\n Digite o segundo numero: ");
```

```
scanf("%d", &n2);
```

```
printf("\n Resultado = %d", n1 + n2);
```

```
return 0;
```

```
}
```

10. Os pares de instruções abaixo produzem o mesmo resultado?

← a.  $A \ (4/2) + (2/4)$  ← e  $A \ 4/2 + 2/4$

← b.  $B \ (4 + 2) * 2 - 4$  ← e  $B \ 4+2*2-4$

A) Ambas produzem o mesmo resultado.

B) Elas produzem resultados diferentes, pois os parênteses mudam a ordem de prioridade.

11. Escreva um algoritmo para ler um valor (do teclado) e escrever (na tela) o seu antecessor.

```
#include <stdio.h>
```

```
int main ()
```

```
{
```

```
int n1, antecessor;
```

```
printf("Insira um numero: ");
```

```
scanf("%d", &n1);
```

```
antecessor = n1-1;
```

```

    printf("\n O Antecessor eh %d", antecessor);

    return 0;
}

```

- 12.** Escreva um algoritmo para ler as dimensões de um retângulo (base e altura), calcular e escrever a área do retângulo.

```

#include <stdio.h>

int main ()
{
    int base, altura;

    printf("\n Base:");
    scanf("%d", &base);
    printf("\n Altura:");
    scanf("%d", &altura);

    printf("\n Area = %d", base*altura);

    return 0;
}

```

- 13.** Escreva um algoritmo para ler um valor e escrever a mensagem “É MAIOR QUE 10!” se o valor lido for maior que 10, ou escrever “NÃO É MAIOR QUE 10!” caso contrário.

```

#include <stdio.h>

int main ()
{
    int numero;

    printf("\n Digite um numero: ");
    scanf("%d", &numero);

    if(numero > 10)
    {
        printf("\n Eh Maior que 10!");
    }
    else
    {
        printf("\n Eh Menor que 10!");
    }

    return 0;
}

```

- 14.** Escreva um algoritmo para ler um valor e escrever a mensagem “É PAR!” se o valor lido for par, ou escrever “É ÍMPAR!” caso contrário.

```
#include <stdio.h>

int main ()
{
    int num1;

    printf("\n Digite Um Numero: ");
    scanf("%d", &num1);

    if(num1 % 2 == 0)
    {
        printf("\n Eh Par!");
    }

    if(num1 % 2 != 0)
    {
        printf("\n Eh Impar!");
    }

    return 0;
}
```

**15.** Escreva um algoritmo para ler o ano atual e o ano de nascimento de uma pessoa.

Escrever uma mensagem que diga se ela poderá ou não votar este ano (não é necessário considerar o mês em que a pessoa nasceu).

```
#include <stdio.h>

int main()
{
    int nascimento, idade;

    printf("\n Em que ano voce nasceu? \n ");
    scanf("%d", &nascimento);

    idade = 2024 - nascimento;

    if(idade >= 16)
    {
        printf("\n Voce pode votar!");
    }

    else
    {
        printf("\n Voce nao pode Votar!");
    }

    return 0;
}
```

- 16.** Escreva um algoritmo para ler dois valores (considere que não serão lidos valores iguais) e escrever o maior deles.

```
#include <stdio.h>

int main ()

{

    int num1, num2;

    printf("\n Digite Um Numero: ");

    scanf("%d", &num1);

    printf("\n Digite Outro Numero: ");

    scanf("%d", &num2);

    if(num1 > num2)

    {

        printf("\n %d eh maior que %d", num1, num2);

    }

    if(num2 > num1)

    {

        printf("\n %d eh maior que %d", num2, num1);

    }

    return 0;

}
```

**Prof. Douglas Francisquini Toledo**

