

09/09/2024

# Eletrônica 4.0

## Módulo 3 – Lista de Exercício 4

INDT - Instituto de Desenvolvimento Tecnológico

Elaborador por: [Diego Machado](#)



6.1. Escreva um programa que leia um ângulo em graus e imprima o seno correspondente. Você pode usar funções padrão.

6.2. Escreva um programa que imprima uma tabela com 2 colunas. A primeira coluna contém todos os ângulos de 0 a 360 graus com passos de 30 graus. A segunda coluna contém os valores de seno correspondentes.

6.3. Escreva um programa que calcule a raiz quadrada de um número inserido pelo usuário.

6.4. Escreva um programa que leia os comprimentos dos lados a e b de um triângulo retângulo e imprima o comprimento da hipotenusa c e um dos ângulos agudos. Exemplos:

Entrada: 2 1

Saída: 2.24 63°

Entrada: 1 1.732

Saída: 2.00 60°

6.5. Escreva um programa para um jogo de adivinhação. Primeiro, um número aleatório entre 1 e 100 é escolhido pelo programa. Depois, o usuário pode começar a adivinhar. Se o palpite for muito alto ou muito baixo, o programa precisa imprimir “muito alto” ou “muito baixo”. Isso é repetido até que o número seja encontrado. No final, o programa imprime quantas tentativas o usuário precisou para encontrar o número secreto.



6.6. Escreva um programa que imprima uma tabela com 2 colunas. A primeira coluna contém valores de  $x$  de -5 a +5 com um passo de 0,5. A segunda coluna contém os valores correspondentes de  $y$  de acordo com a equação  $y = 2x^2 + 2x - 3$ . Certifique-se de que o cálculo dos valores de  $y$  seja feito em uma função separada.

6.7. Escreva uma função com o cabeçalho:

```
void printline(int number, char c)
```

Exemplo:

Entrada

```
printline(40, '*');
```

Saida

```
*****
```

6.8 Escreva uma função que leia um número inteiro no intervalo  $[0, 10]$  e retorne esse número como valor de retorno da função. Se o número não estiver no intervalo correto, a função deve pedir um novo número até que um valor correto seja inserido. O programa principal é algo como:



6.6. Escreva um programa que imprima uma tabela com 2 colunas. A primeira coluna contém valores de  $x$  de -5 a +5 com um passo de 0,5. A segunda coluna contém os valores correspondentes de  $y$  de acordo com a equação  $y = 2x^2 + 2x - 3$ . Certifique-se de que o cálculo dos valores de  $y$  seja feito em uma função separada.

6.7. Escreva uma função com o cabeçalho:

```
void printline(int number, char c)
```

Exemplo:

Entrada

```
printline(40, '*');
```

Saida

```
*****
```



6.8 Escreva uma função que leia um número inteiro no intervalo [0, 10] e retorne esse número como valor de retorno da função. Se o número não estiver no intervalo correto, a função deve pedir um novo número até que um valor correto seja inserido. O programa principal é algo como:

```
int main(void)
{
    int number;
    number = readnumber();
    printf("O número lido é %d\n", number);
    return 0;
}
```

Certifique-se de escrever a declaração da função `readnumber()` antes da função principal e a definição da função após a função principal.

6.9. Escreva uma função com o cabeçalho:

```
int readnumber(int lower_boundary, int upper_boundary)
```

que lê um número inteiro no intervalo [lower\_boundary, upper\_boundary] e retorna esse número como valor de retorno da função. Também neste caso, a função só pode parar de pedir um número inteiro se um valor correto for inserido. Escreva uma função principal que chame essa função e imprima o número resultante.



6.10. Escreva uma função com o cabeçalho:

```
double exponentiation(double base, int exponent)
```

que retorna  $\text{base}^{\text{exponent}}$  como valor de retorno da função. Escreva uma função principal que leia uma base e um expoente, chame a função `exponentiation` e imprima o resultado. Não use a função padrão `pow`!

6.11. Escreva uma função com o cabeçalho:

```
int gcd(int number1, int number2)
```

que retorna o maior divisor comum dos números `number1` e `number2` como valor de retorno da função. Escreva um programa principal que leia 3 números inteiros, calcule o MDC (máximo divisor comum) desses 3 números e imprima o resultado. Dica:  $\text{gcd}(a, b, c) = \text{gcd}(a, \text{gcd}(b, c))$



6.12. Escreva um programa que leia uma quantidade de pontuações. As pontuações são todos números inteiros positivos. Um número negativo é inserido para indicar que todas as pontuações foram inseridas. Para cada pontuação, uma barra com um comprimento igual à pontuação é desenhada. Para isso, um símbolo pré-definido precisa ser impresso tantas vezes quanto a pontuação. (uso de laço for).

Escreva uma função `draw_bar` que recebe uma pontuação como entrada e desenha uma barra com o comprimento correspondente.

As pontuações e o símbolo desejado são lidos na função principal.

O diálogo na tela deve ser como:

Digite as pontuações: 2 12 18 3 -5  
Que símbolo você gostaria de usar? =

```
pontuação = 2 ==  
pontuação = 12 =====  
pontuação = 18 =====  
pontuação = 3 ===
```



6.13. Escreva um programa com as funções:

- hello: que dá as boas-vindas ao usuário e explica o que é esperado
- main: que pede ao usuário para inserir 5 vezes 2 números
- sum: que pede ao usuário para inserir a soma dos números anteriormente inseridos e dá um feedback ao usuário
- goodbye: que agradece ao usuário pela cooperação Bem-vindo, este programa pedirá que você resolva 5 somas. digite 2 números < 100: 15 16 qual é a soma de 15 e 16? 31 segundo você, a soma de 15 e 16 é 31. Isso está correto digite 2 números < 100: 26 32 qual é a soma de 26 e 32? 50 segundo você, a soma de 26 e 32 é 50. Isso está incorreto ... Obrigado pela sua cooperação





6.14. Como um ano não tem exatamente 365 dias, temos um ano bissexto a cada 4 anos, exceto se o ano for divisível por 100. Se o ano for divisível por 400, o ano é considerado bissexto mesmo assim. Escreva uma função com o cabeçalho:

```
int isLeapYear(int year)
```

que determina se um ano é bissexto ou não e dá um valor de retorno diferente para ambos os casos. Escreva também uma função com o cabeçalho:

```
int numberOfDays(int month, int year)
```

que calcula o número de dias no mês month do ano dado. Escreva uma função principal que leia um mês e um ano e imprima o número de dias nesse mês e ano. Exemplos: mês 2 de 2000 tem 29 dias mês 2 de 1900 tem 28 dias mês 1 de 1950 tem 31 dias

6.15. Escreva uma função com 3 números inteiros que representam um dia, mês e ano. Essa função calcula e retorna um fator de acordo com a seguinte fórmula:

Para os meses de Janeiro e Fevereiro:

$$\text{fator} = 365 * \text{year} + \text{day} + 31 * (\text{month} - 1) + [\text{year} - 1 / 4] - [[\text{year} - 1 / 100 + 1] * 3 / 4]$$

Para os meses de Março a Dezembro:

$$\text{fator} = 365 * \text{year} + \text{day} + 31 * (\text{month} - 1) - [0.4 * \text{month} + 2.3] + [\text{year} / 4] - [[\text{year} / 100 + 1] * 3 / 4]$$

Escreva uma função principal que leia 2 datas, calcule o número de dias entre essas 2 datas calculando a diferença entre os 2 fatores das datas correspondentes.



6.16. Considere o primeiro quadrante de um círculo em um quadrado com lado 1. Se você gerar uma grande quantidade de coordenadas  $(x, y)$  com  $x$  e  $y$  pertencendo ao intervalo  $[0, 1]$ , você terá uma coleção de pontos pertencentes ao quadrado. Se você contar todos os pontos que pertencem ao quadrante do círculo com equação  $x^2 + y^2 < 1$  e dividir essa quantidade pelo número total de pontos gerados, você encontrará aproximadamente o número  $\pi/4$ .

- Escreva um programa que gera 100.000 pontos e usa isso para calcular o número  $\pi$  aproximadamente.
- Execute o programa várias vezes e compare os resultados.
- E se você aumentar o número de pontos?



6.17. Escreva um programa que peça ao usuário para escolher entre 3 opções:

1. Desenhar um quadrado vazio

2. Desenhar um quadrado preenchido

3. Parar O programa se repete até que o usuário escolha a opção 3 (Parar). Use as funções `empty_square` e `filled_square`.

```
What do you choose?
  1. Draw an empty square
  2. Draw a filled square
  3. Stop
5
Enter a valid choice: 1, 2 or 3!
What do you choose?
  1. Draw an empty square
  2. Draw a filled square
  3. Stop
1
[Empty square]
What do you choose?
  1. Draw an empty square
  2. Draw a filled square
  3. Stop
2
[Filled square]
What do you choose?
  1. Draw an empty square
  2. Draw a filled square
  3. Stop
3
Thanks for playing with us!
```



 @indt.instituto

 [linkedin.com/company/indt-instituto](https://www.linkedin.com/company/indt-instituto)

[www.indt.org.br](http://www.indt.org.br)