



LISTA DE EXERCÍCIOS – ESTRUTURAS DE REPETIÇÃO

ATENÇÃO:

Para esta lista use apenas expressões, desvios condicionais e estruturas de repetição.  
 Seu programa não deve fazer uso de variáveis que não sejam de tipo básico.

- Suponha que seu computador consiga executar somente operações de soma e subtração. Escreva programas que, dados dois números  $a$  e  $b$ , não necessariamente positivos, calculem:
  - O produto  $ab$ .
  - O quociente e o resto da divisão de  $a$  por  $b$ .
- Escreva um programa que leia um número inteiro positivo  $n$ , em seguida leia  $n$  números inteiros e calcule a média destes números, o maior e o menor deles.
- Escreva um programa que leia um inteiro não negativo  $n$ , em seguida leia  $n$  dígitos e calcule a frequência de ocorrência de cada um deles.
- Escreva um programa que leia um inteiro não negativo  $n$ , em seguida leia  $n$  letras. Seu programa deve retornar a frequência de ocorrência de cada letra.
- Escreva um programa que leia um número inteiro  $n$  e imprima uma tabela com  $n$  linhas, tal que a  $i$ -ésima linha contenha:

$$i \quad i^2 \quad i^3$$

- Escreva um programa que leia dois números  $m$  e  $n$  e imprima uma tabela de multiplicação com  $m$  linhas e  $n$  colunas de maneira que a  $i$ -ésima linha seja:

$$1 * i \quad 2 * i \quad 3 * i \quad \dots \quad (m - 2) * i \quad (m - 1) * i \quad m * i$$

- Suponha que voce invista seu dinheiro a juros fixos de  $r\%$  ao mês. Após  $n$  meses, o seu investimento crescerá segundo a seguinte fórmula:

Número de Meses	Investimento Acumulado
1	$a + (r \times a) = a(1 + r)$
2	$a(1 + r) \times (1 + r) = a(1 + r)^2$
3	$a(1 + r)^2 \times (1 + r) = a(1 + r)^3$
$\vdots$	$\vdots$
$n$	$a(1 + r)^{n-1} \times (1 + r) = a(1 + r)^n$

Escreva um programa para calcular e escrever a tabela acima, dado um investimento inicial  $a$ , um número  $n$  de meses e juros de  $r\%$ .

- Escreva um programa que leia um número inteiro  $n$  e escreva uma figura similar à seguinte, mas com  $n$  linhas.

```
...*. . . . .*. . .
. .***. . . .***. .
.*****. .*****.
*****
```

- Escreva um programa que leia um número inteiro positivo  $n$  e em seguida imprima  $n$  linhas do chamado *triângulo de Floyd*. O exemplo abaixo mostra o triângulo de Floyd com 6 linhas.

```

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21

```

10. Escreva um programa que leia um inteiro positivo  $n$  e imprima um triângulo de  $n$  linhas, constituído por números com o seguinte formato:

```

(Para n=4)
4 3 2 1
3 2 1
2 1
1

```

11. Dado um número inteiro não negativo  $n$ , seja  $reverso(n)$  o número que se obtém invertendo-se a ordem dos dígitos de  $n$ . Por exemplo,  $reverso(332)=233$ . Um número é um *palíndromo* se  $reverso(n)=n$ . Por exemplo, 34543, 1, 99 são palíndromos. Escreva um programa que leia um número  $n$  e verifique se  $n$  é um palíndromo, imprimindo a resposta adequada.
12. Escreva o que seria impresso pelo programa abaixo se o valor lido na variável  $a$  fosse o número inteiro correspondente aos dois últimos dígitos do seu RA.

```

#include <stdio.h>

int main()
{
    int a, s, i;
    scanf("%d",&a);

    i = 19;
    s = 100;
    while (s>a) {
        s = s - i;
        i = i - 2;
    }

    printf("%d",s);

    return 0;
}

```

13. Modifique o programa anterior para usar **do-while** ao invés de **while**.
14. Seja  $S$  um texto formado por letras maiúsculas, vírgulas, pontos e brancos, terminado pelo caracter ‘#’ (que somente ocorre no fim do texto). Escreva um programa que leia os caracteres de  $S$  um por vez e imprima o número de palavras com comprimento menor ou igual a 5.
15. Faça um programa que leia um número inteiro positivo  $n$ , em seguida leia  $n$  pares de números  $p_i, x_i$ , calcule e imprima a média ponderada destes números de acordo com a fórmula:

$$\frac{\sum_{i=1}^n p_i * x_i}{\sum_{i=1}^n p_i}$$

16. Faça um programa que calcule a aproximação para a integral:

$$\int_0^x e^{-u^2} du = x - \frac{x^3}{3 * 1!} + \frac{x^5}{5 * 2!} - \frac{x^7}{7 * 3!} + \frac{x^9}{9 * 4!} \dots$$

O seu programa deve calcular  $n$  termos da aproximação, onde  $n$  é um valor de entrada.

17. Escreva um programa que leia um número inteiro positivo e imprima a representação deste número em algarismos romanos.