



Lista 3 - Funções

Exercícios para sala de aula - Revisão

1) Fazer funções para:

- a) Mostrar os múltiplos de y entre 1 (se for o caso) e x , sendo x o valor até o qual os múltiplos de y devem ser mostrados. A função recebe por parâmetro o número e valor que representa "múltiplo de". Por exemplo: parâmetros 10 (como x) e 2 (como y) significa os múltiplos de 2 entre 1 e 10.
- b) Somar os múltiplos de y entre 1 (se for o caso) e x , sendo x o valor até o qual os múltiplos devem ser somados. Por exemplo: parâmetros 10 (como x) e 2 (como y) significa somar os múltiplos de 2 entre 1 e 10. O resultado da soma deve ser retornado da função.

Fazer um programa que:

- a) Leia dois valores sendo que o primeiro representa o limite final do intervalo (o limite inicial é 1) e o segundo valor representa "múltiplo de".
- b) Leia três valores, sendo que o primeiro e o segundo representam, respectivamente, o limite inicial e final de um intervalo e o terceiro valor representa "múltiplo de". Utilize as duas funções, criadas anteriormente, para apresentar a saída da forma como apresentado a seguir. Nesse exemplo os valores para os limites inferior e superior são 10 e 13 e o "múltiplo de" é 2.

10 = 2 4 6 8 10 => Soma: 30

11 = 2 4 6 8 10 => Soma: 30

12 = 2 4 6 8 10 12 => Soma: 42

13 = 2 4 6 8 10 12 => Soma: 42

Exemplo:

1 - Mostra e soma os multiplos de um numero
2 - Mostra e soma os multiplos dos numeros de um intervalo
Opcao: 1

Informe um numero: 12
Mostrar todos os multiplos de: 3
12 = 3 6 9 12 => Soma: 30

Deseja repetir o programa (S ou N)? s
1 - Mostra e soma os multiplos de um numero
2 - Mostra e soma os multiplos dos numeros de um intervalo
Opcao: 2

Informe o valor do limite inferior de um intervalo: 10
Informe o valor do limite superior de um intervalo: 13
Para cada numero do intervalo, mostrar todos os multiplos de: 2
10 = 2 4 6 8 10 => Soma: 30
11 = 2 4 6 8 10 => Soma: 30
12 = 2 4 6 8 10 12 => Soma: 42
13 = 2 4 6 8 10 12 => Soma: 42

Deseja repetir o programa (S ou N)? n

2) Fazer uma função para verificar se um número equivale a soma dos n primeiros ímpares iniciando em 1. Por exemplo, 9, é um número que tem essa característica porque $1 + 3 + 5 = 9$ (soma dos n primeiros números ímpares iniciando em 1). Essa função recebe como parâmetro um valor inteiro e retorna 's' se o número equivale a soma dos n primeiros ímpares e 'n' caso não.

Fazer um programa que use essa função para:

- Ler um número inteiro informado pelo usuário e verificar se o mesmo tem essa característica.
- Mostrar todos os números que têm ou não têm essa característica e pertencem a um intervalo informado pelo usuário.

Exemplo:

```
1 - Verifica se um numero equivale a soma dos impares
2 - Verifica os numeros que equivalem a soma dos impares de um intervalo
Opcao: 1
```

Informe um numero: 9

$9 = 1 + 3 + 5 \Rightarrow$ Equivale a soma dos ímpares

Deseja repetir o programa (S ou N)? s

```
1 - Verifica se um numero equivale a soma dos impares
2 - Verifica os numeros que equivalem a soma dos impares de um intervalo
Opcao: 1
```

Informe um numero: 5

$5 = 1 + 3 + 5 \Rightarrow$ Não equivale a soma dos ímpares

Deseja repetir o programa (S ou N)? s

```
1 - Verifica se um numero equivale a soma dos impares
2 - Verifica os numeros que equivalem a soma dos impares de um intervalo
Opcao: 2
```

```
Informe o valor do limite inferior de um intervalo: 1
Informe o valor do limite superior de um intervalo: 10
```

```

1 = 1 => Equivale a soma dos impares
2 = 1 => Nao equivale a soma dos impares
3 = 1 3 => Nao equivale a soma dos impares
4 = 1 3 => Equivale a soma dos impares
5 = 1 3 5 => Nao equivale a soma dos impares
6 = 1 3 5 => Nao equivale a soma dos impares
7 = 1 3 5 => Nao equivale a soma dos impares
8 = 1 3 5 => Nao equivale a soma dos impares
9 = 1 3 5 => Equivale a soma dos impares
10 = 1 3 5 7 => Nao equivale a soma dos impares

```

Deseja repetir o programa (S ou N)?

3) (DESAFIO) Fazer um programa que receba um valor inteiro, que se refere a quantidade de linhas de um triângulo. Implementar no próprio programa uma função que recebe como parâmetro esse valor e imprime um triângulo.

Exemplo:

Informe a quantidade de linhas do triangulo: 10

```

      *
    ***
  *****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****

```

Deseja repetir o programa (S ou N)? n

4) (DESAFIO) Incremente o código da função do exercício anterior e transforme o triângulo em uma árvore de Natal, como mostrado no exemplo a seguir.

Exemplo:

Insira a altura(em linhas) da arvore: 10

```
  *
 ***
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
  *
  *
  *
  *
  *
```

Deseja repetir o programa (S ou N)? n

5) (DESAFIO) Fazer um programa que receba um valor inteiro, que se refere a quantidade de linhas de um triângulo e um valor do tipo *char*, que se refere ao tipo do triângulo (que pode ser preenchido ou vazado). Implementar no próprio programa uma função que recebe como parâmetros esses valores e imprime um triângulo como no exemplo a seguir.

Exemplo:

Informe a altura do triangulo em linhas: 10
Vazado(V/v) ou Preenchido(P/p)? p

```
*
**
***
****
*****
*****
*****
*****
*****
*****
*****
```

Deseja repetir o programa (S ou N)? s

Informe a altura do triangulo em linhas: 10
Vazado(V/v) ou Preenchido(P/p)? v

```
*
**
* *
*  *
*   *
*    *
*     *
*      *
*       *
*        *
*****
```

Deseja repetir o programa (S ou N)? n

6) Para saber se um número é feliz, você deve obter o quadrado de cada dígito deste número, em seguida você faz a soma desses resultados. A seguir o mesmo procedimento deve ser feito com o valor resultante desta soma. Se ao repetir o procedimento diversas vezes obtivermos o valor 1, o número inicial é considerado feliz.

Tomamos o 7, que é um número feliz:

$$7^2 = 49$$

$$4^2 + 9^2 = 97$$

$$9^2 + 7^2 = 130$$

$$1^2 + 3^2 + 0^2 = 10$$

$$1^2 + 0^2 = 1$$

Podemos observar nesse exemplo que os números 49, 97, 130 e 10 também são felizes. Existem infinitos números felizes.

E um número triste? Como sabemos que um número não é feliz?

Desenvolva uma função que receba um número inteiro positivo e retorne 1 se o número for feliz e 0 se o número não for feliz. Elabore um programa que use essa função para determinar se um número é ou não feliz.

7) Use a função criada no exercício anterior para mostrar todos os números felizes existentes no intervalo de 1 até 500.

8) (DESAFIO) Desenvolva uma função que receba um número inteiro positivo e ímpar e imprima um diamante de asteriscos. Escreva um programa para validar a entrada e chamar a função.

Exemplo:

Insira um numero impar: 13

```

      *
    ***
  *****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****

```

Deseja repetir o programa (S ou N)?