

01. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o fatorial de n utilizando a fórmula

$$f = 1 * 2 * 3 * 4 * \dots * n$$

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor fornecido pelo usuário para n seja 5, o programa deverá apresentar como resposta o valor 120, ou seja, $1 * 2 * 3 * 4 * 5$.

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

02. Desenvolver um programa recursivo, sobre uma máquina genérica, que estime o valor da constante matemática e utilizando a fórmula

$$e = 1/0! + 1/1! + 1/2! + 1/3! + \dots + 1/n!$$

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor fornecido pelo usuário para n seja 5, o programa deverá apresentar como resposta o valor 2.72, ou seja, $1/0! + 1/1! + 1/2! + 1/3! + 1/4! + 1/5!$.

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

03. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o valor de e^x utilizando a fórmula

$$e^x = x^0/0! + x^1/1! + x^2/2! + x^3/3! + \dots + x^n/n!$$

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

O valor de x será fornecido pelo usuário, podendo ser um valor (inteiro ou real) qualquer.

Por exemplo, caso o valor fornecido pelo usuário para n seja 4 e para x seja 2, o programa deverá apresentar como resposta o valor 7, ou seja, $2^0/0! + 2^1/1! + 2^2/2! + 2^3/3! + 2^4/4!$.

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

04. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o valor de π com a série infinita

$$\pi = 4 - 4/3 + 4/5 - 4/7 + 4/9 - 4/11 + \dots$$

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta o valor 3.34, ou seja, $4 - 4/3 + 4/5 - 4/7 + 4/9$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

05. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o valor da exponenciação de uma base qualquer por um expoente positivo qualquer, ou seja, B^E .

O valor de E será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

O valor de B será fornecido pelo usuário, podendo ser um valor (inteiro ou real) qualquer.

Por exemplo, caso o valor fornecido pelo usuário para B seja 2 e para E seja 5, o programa deverá apresentar como resposta o valor 32, ou seja, $2 * 2 * 2 * 2 * 2$.

Caso o usuário forneça um valor inválido para E , o programa deverá apresentar uma mensagem de erro.

06. Desenvolver um programa recursivo, sobre uma máquina genérica, que leia 100 (cem) valores inteiros (positivos ou negativos) fornecidos pelo usuário, e apresente ao final da leitura o maior e o menor valor fornecido pelo usuário.

07. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o valor da série infinita

$$S = 1 + 1/2 + 1/4 + 1/6 + 1/8 + 1/10 + \dots$$

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta o valor 2.04, ou seja, $1 + 1/2 + 1/4 + 1/6 + 1/8$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

08. Desenvolver um programa recursivo, sobre uma máquina genérica, que retorne a somatória de n utilizando a fórmula

$$S = 0 + 1 + 2 + 3 + \dots + n$$

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor fornecido pelo usuário para n seja 5, o programa deverá apresentar como resposta o valor 15, ou seja, $0 + 1 + 2 + 3 + 4 + 5$.

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

09. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o valor da série infinita

$$H = 1^1/1! - 2^2/2! + 3^3/3! - 4^4/4! + \dots$$

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta o valor 18.88, ou seja, $1^1/1! - 2^2/2! + 3^3/3! - 4^4/4! + 5^5/5!$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

10. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente a série de *Fibonacci*. A série de *Fibonacci* é formada pela sequência

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

A série de *Fibonacci* é de grande importância matemática, e a lei básica é que a partir do terceiro termo, todos os termos são a soma dos dois últimos.

O número de termos a serem impressos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos a serem impressos fornecido pelo usuário seja 7, o programa deverá apresentar como resposta a sequência de valores 1, 1, 2, 3, 5, 8, 13.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

11. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente a somatória dos termos da série de *Fibonacci*. A série de *Fibonacci* é formada pela sequência

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

A série de *Fibonacci* é de grande importância matemática, e a lei básica é que a partir do terceiro termo, todos os termos são a soma dos dois últimos.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 7, o programa deverá apresentar como resposta o valor 33, ou seja, $1 + 1 + 2 + 3 + 5 + 8 + 13$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

12. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente o produto dos termos da série de *Fibonacci*. A série de *Fibonacci* é formada pela sequência

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

A série de *Fibonacci* é de grande importância matemática, e a lei básica é que a partir do terceiro termo, todos os termos são a soma dos dois últimos.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 7, o programa deverá apresentar como resposta o valor 3120, ou seja, $1 * 1 * 2 * 3 * 5 * 8 * 13$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

13. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente o valor da série infinita

$$R = 1/1 + 2/1 + 3/2 + 4/3 + 5/5 + 6/8 + 7/13 + 8/21 + 9/34 + \dots$$

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 6, o programa deverá apresentar como resposta o valor 7.58, ou seja, $1/1 + 2/1 + 3/2 + 4/3 + 5/5 + 6/8$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

14. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente o valor da série infinita

$$G = 1/1! - 2/1! + 3/2! - 4/3! + 5/5! - 6/8! + \dots$$

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta o valor -0.13, ou seja, $1/1! - 2/1! + 3/2! - 4/3! + 5/5!$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

15. Desenvolver um programa recursivo, sobre uma máquina genérica, que verifique se o valor inteiro e positivo fornecido pelo usuário é um número primo ou não.

Um número primo é um número que é divisível apenas por 1 e por ele mesmo.

Por exemplo, caso o valor fornecido pelo usuário seja 5, o programa deverá apresentar como resposta `É primo`.

Por exemplo, caso o valor fornecido pelo usuário seja 8, o programa deverá apresentar como resposta `Não é primo`.

Caso o usuário forneça um valor inválido, o programa deverá apresentar uma mensagem de erro.

16. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente os quadrados dos números inteiros de 1 a n .

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor fornecido pelo usuário para n seja 5, o programa deverá apresentar como resposta a sequência de valores 1, 4, 9, 16, 25.

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

17. Desenvolver um programa recursivo, sobre uma máquina genérica, que escreva simultaneamente a sequência crescente e decrescente entre 1 e n .

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor fornecido pelo usuário para n seja 5, o programa deverá apresentar como resposta a sequência de valores

```
1 5
2 4
3 3
4 2
5 1
```

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

18. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente os n primeiros termos da sequência $y_{k+1} = y_k + 2$, sendo $k = 1, 2, \dots, n$ e $y_1 = 1$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta a sequência de valores 1, 3, 5, 7, 9

```
y1 = 1
y2 = y1 + 2 = 3
y3 = y2 + 2 = 5
y4 = y3 + 2 = 7
y5 = y4 + 2 = 9
```

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

19. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente o valor aproximado da raiz quadrada de um número A , por meio de n iterações, através da sequência de aproximação $x_n = (x_{n-1} + A/x_{n-1}) / 2$, com $x_1 = 1$ e $n \in \mathbb{N}$.

O número de iterações e o valor de A serão fornecidos pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor fornecido pelo usuário para o número de iterações seja 5 e para A seja 3, o programa deverá apresentar como resposta o valor 1.732050810, obtido pela sequência de valores

```
x1 = 1
x2 = (x1 + 3/x1) / 2 = 2
x3 = (x2 + 3/x2) / 2 = 1.75
x4 = (x3 + 3/x3) / 2 = 1.732142857
x5 = (x4 + 3/x4) / 2 = 1.732050810
```

Caso o usuário forneça um valor inválido para o número de iterações ou para A , o programa deverá apresentar uma mensagem de erro.

20. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente os n primeiros termos da sequência $y_k = y_{k-1} + k$, sendo $k = 1, 2, \dots, n$ e $y_1 = 1$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta a sequência de valores 1, 3, 6, 10, 15.

$$\begin{aligned}y_1 &= 1 \\y_2 &= y_1 + 2 = 3 \\y_3 &= y_2 + 3 = 6 \\y_4 &= y_3 + 4 = 10 \\y_5 &= y_4 + 5 = 15\end{aligned}$$

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

21. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente os n primeiros termos da sequência $y_{k+1} = y_k + (2k + 1)$, sendo $k = 1, 2, \dots, n$ e $y_1 = 1$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta a sequência de valores 1, 4, 9, 16, 25.

$$\begin{aligned}y_1 &= 1 \\y_2 &= y_1 + (2 * 1 + 1) = 4 \\y_3 &= y_2 + (2 * 2 + 1) = 9 \\y_4 &= y_3 + (2 * 3 + 1) = 16 \\y_5 &= y_4 + (2 * 4 + 1) = 25\end{aligned}$$

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

22. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente os n primeiros termos da sequência $y_{k+1} = y_k + (3k^2 + 3k + 1)$, sendo $k = 1, 2, 3, \dots, n$ e $y_1 = 1$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta a sequência de valores 1, 8, 27, 64, 125.

$$\begin{aligned}y_1 &= 1 \\y_2 &= y_1 + (3 * 1^2 + 3 * 1 + 1) = 8 \\y_3 &= y_2 + (3 * 2^2 + 3 * 2 + 1) = 27 \\y_4 &= y_3 + (3 * 3^2 + 3 * 3 + 1) = 64 \\y_5 &= y_4 + (3 * 4^2 + 3 * 4 + 1) = 125\end{aligned}$$

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

- 23.** Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente os n primeiros termos da sequência $y_{k+1} = 2y_k$, sendo $k = 1, 2, \dots, n$ e $y_1 = 1$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta a sequência de valores 1, 2, 4, 8, 16.

$$\begin{aligned}y_1 &= 1 \\y_2 &= 2 * y_1 = 2 \\y_3 &= 2 * y_2 = 4 \\y_4 &= 2 * y_3 = 8 \\y_5 &= 2 * y_4 = 16\end{aligned}$$

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

- 24.** Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente a somatória de $2k$, sendo $k = 1, 2, \dots, n$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta o valor 30, ou seja, $2 * 1 + 2 * 2 + 2 * 3 + 2 * 4 + 2 * 5$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

- 25.** Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente a somatória de k^2 , sendo $k = 1, 2, \dots, n$.

O número de termos será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o número de termos fornecido pelo usuário seja 5, o programa deverá apresentar como resposta o valor 55, ou seja, $1^2 + 2^2 + 3^2 + 4^2 + 5^2$.

Caso o usuário forneça um valor inválido para o número de termos, o programa deverá apresentar uma mensagem de erro.

- 26.** Desenvolver um programa recursivo, sobre uma máquina genérica, que leia três números do usuário e apresente o maior número fornecido.

Por exemplo, caso os números fornecidos pelo usuário sejam 5, 7 e 4, o programa deverá apresentar como resposta o valor 7, que é o maior número entre os três números fornecidos pelo usuário.

- 27.** Desenvolver um programa recursivo, sobre uma máquina genérica, que leia três números do usuário e apresente o menor número fornecido.

Por exemplo, caso os números fornecidos pelo usuário sejam 5, 7 e 4, o programa deverá apresentar como resposta o valor 4, que é o menor número entre os três números fornecidos pelo usuário.

28. Desenvolver um programa recursivo, sobre uma máquina genérica, que leia três números do usuário e os apresente em ordem crescente.

Por exemplo, caso os números fornecidos pelo usuário sejam 5, 7 e 4, o programa deverá apresentar como resposta a sequência 4, 5 e 7.

29. Desenvolver um programa recursivo, sobre uma máquina genérica, que leia três números do usuário e os apresente em ordem decrescente.

Por exemplo, caso os números fornecidos pelo usuário sejam 5, 7 e 4, o programa deverá apresentar como resposta a sequência 7, 5 e 4.

30. Desenvolver um programa recursivo, sobre uma máquina genérica, que apresente a somatória dos n primeiros números pares positivos.

O valor de n será fornecido pelo usuário, devendo ser um valor inteiro e positivo.

Por exemplo, caso o valor de n fornecido pelo usuário seja 8, o programa deverá apresentar como resposta o valor 20, ou seja, $2 + 4 + 6 + 8$.

Caso o usuário forneça um valor inválido para n , o programa deverá apresentar uma mensagem de erro.

31. Desenvolver um programa recursivo, sobre uma máquina genérica, que calcule o quadrado dos números fornecidos pelo usuário, até que o mesmo forneça um número negativo para ser calculado.

Por exemplo, caso os valores fornecidos pelo usuário sejam 1 7 4 3 -6, o programa deverá apresentar como resposta os valores 1 49 16 9, ou seja, 1^2 7^2 4^2 3^2 .

32. Desenvolver um programa recursivo, sobre uma máquina genérica, que dado dois números inteiros positivos, determine quantas vezes o primeiro divide exatamente o segundo.

Se o primeiro número não divide o segundo, o número de vezes é zero.

Os valores dos dois números serão fornecidos pelo usuário, devendo ser valores inteiros e positivos.

Por exemplo, caso os valores fornecidos pelo usuário sejam 2 e 8, o programa deverá apresentar como resposta o valor 3, ou seja, $8 / 2 = 4 / 2 = 2 / 2 = 1$.

Caso o usuário forneça um valor inválido para alguns dos dois números, o programa deverá apresentar uma mensagem de erro.