**EXERCÍCIO - Recursão**

1 - Escreva um procedimento recursivo que recebe como entrada um número K do tipo inteiro. O procedimento recursivo escreve no console o fatorial de K.

|  |  |
| --- | --- |
| **ENTRADA** | **SAÍDA** |
| 3 | 6 |
| 6 | 720 |
| 0 | 1 |

2 - Escreva um procedimento recursivo que recebe como entrada um número K do tipo inteiro. O procedimento recursivo escreve no console o número de Fibonacci de K.

|  |  |
| --- | --- |
| **ENTRADA** | **SAÍDA** |
| 3 | 2 |
| 6 | 8 |
| 0 | 0 |

3 - Escreva um procedimento recursivo que recebe como entrada um número K do tipo inteiro. O procedimento recursivo escreve no console todos os números n, onde n é um número natural e menor ou igual a K.

|  |  |
| --- | --- |
| **ENTRADA** | **SAÍDA** |
| 3 | 1  2  3 |
| 2 | 1  2 |
| 0 | -1 |

4 - Escreva um procedimento recursivo que recebe como entrada um número K do tipo inteiro. O procedimento recursivo escreve no console todos os números n, onde n é um número natural e menor ou igual a K, de forma decrescente e, depois, crescente.

|  |  |
| --- | --- |
| **ENTRADA** | **SAÍDA** |
| 3 | 3 2 1 1 2 3 |
| 5 | 5 4 3 2 1 1 2 3 4 5 |
| 8 | 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7 8 |

5 - Escreva um procedimento recursivo que recebe como entrada uma mensagem do tipo caractere e um número K do tipo inteiro. O procedimento recursivo escreve no console a mensagem passada K vezes.

|  |  |  |
| --- | --- | --- |
| **ENTRADA** | |  |
| **String** | **K** | **SAÍDA** |
| Olá Mundo! | 3 | Olá Mundo!  Olá Mundo!  Olá Mundo! |
| Testando | 1 | Testando |
| Adeus | 0 |  |

6 - Escreva uma função recursiva para calcular o produto de dois números inteiros. Ela receberá como entrada os dois números e retorna como resultado o produto deles. Crie também um programa para testar essa função que você escreveu (dica: tente representar o produto como uma soma).

|  |  |  |
| --- | --- | --- |
| **ENTRADA** | |  |
| **Primeiro número** | **Segundo número** | **SAÍDA** |
| 4 | 6 | 24 |
| 1 | 11 | 11 |
| 0 | 6 | 0 |

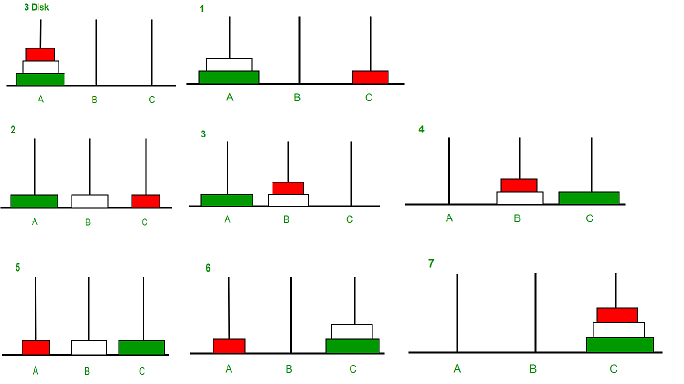
7 - Escreva uma função recursiva que receba um número inteiro e retorne a quantidade de dígitos deste número. Crie um programa para testar essa função, onde você lerá um número fornecido pelo usuário e, sem seguida, informará a quantidade de dígitos desse número.

|  |  |
| --- | --- |
| **ENTRADA** | **SAÍDA** |
| 9 | 1 |
| 154 | 3 |

8 - Escreva uma função recursiva que retorne o menor elemento de um vetor com 10 elementos. Para isso, ela receberá como entrada o vetor e o número de elementos no mesmo. A função retornará o menor número do vetor.

|  |  |  |
| --- | --- | --- |
| **ENTRADA** | |  |
| **Número de elementos** | **Vetor** | **SAÍDA** |
| 10 | <4,6,4,3,2,8,9,5,2,6> | 2 |

9 - Escreva uma função recursiva que resolve o problema da Torre de Hanoi. um "quebra-cabeça" que consiste em uma base contendo três pinos, em um dos quais são dispostos alguns discos uns sobre os outros, em ordem crescente de diâmetro, de cima para baixo. O problema consiste em passar todos os discos de um pino para outro qualquer, usando um dos pinos como auxiliar, de maneira que um disco maior nunca fique em cima de outro menor em nenhuma situação. (Para quem quiser brincar um pouco a Torre de Hanoi: https://www.mathsisfun.com/games/towerofhanoi.html)



|  |  |
| --- | --- |
| **ENTRADA** | **SAÍDA** |
| 4 | Mover disco 1 de SOURCE para AUX  Mover disco 2 de SOURCE para DESTINATION  Mover disco 1 de AUX para DESTINATION  Mover disco 3 de SOURCE para AUX  Mover disco 1 de DESTINATION para SOURCE  Mover disco 2 de DESTINATION para AUX  Mover disco 1 de SOURCE para AUX  Mover disco 4 de SOURCE para DESTINATION  Mover disco 1 de AUX para DESTINATION  Mover disco 2 de AUX para SOURCE  Mover disco 1 de DESTINATION para SOURCE  Mover disco 3 de AUX para DESTINATION  Mover disco 1 de SOURCE para AUX  Mover disco 2 de SOURCE para DESTINATION  Mover disco 1 de AUX para DESTINATION |

10 – Escreva uma função recursiva que recebe como entrada duas strings A e B e conta o número de ocorrências de B em A. Casos de teste são apresentados abaixo.

|  |  |  |
| --- | --- | --- |
| **ENTRADAS** | | **SAÍDA** |
| **String (A)** | **Substring (B)** | **Count** |
| aaadsaa | aa | 3 |
| aaifpbaa | aa | 2 |
| aaifpbaa | ifpb | 1 |
| ifpbaa | aa | 1 |
| aaaaa | aa | 4 |