

# Lista de Exercícios de Recursão

1. Faça uma função que, recebendo como parâmetro um único número inteiro, resulta verdadeiro, caso o mesmo seja negativo, ou falso, caso contrário.

Na implementação de sua função, fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de qualquer função de biblioteca ou de sua própria autoria (fica liberado o uso de funções auxiliares que seguem as mesmas regras aqui impostas e o uso de funções implementadas nesta lista de exercícios em outras funções da mesma lista de exercícios).

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

**Daqui para cima, é admissível fazer uma função auxiliar com o único propósito de auxiliar; daqui para baixo, só se pode usar como auxiliar uma outra função pedida nesta lista de exercícios.**

2. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta a soma deles.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

3. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta a subtração deles (o primeiro menos o segundo).

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

4. Faça uma única função que, recebendo como parâmetro um número inteiro, resulta módulo (valor absoluto) dele.

Importante lembrar que o número com o qual vocês trabalharão é inteiro, logo, pode ser positivo ou negativo.

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

5. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta a multiplicação deles.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

6. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta a potenciação deles, ou seja, o primeiro deles elevado ao segundo deles.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

7. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta a divisão inteira deles (o primeiro dividido pelo segundo).

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

8. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta o resto da divisão inteira deles (o resto da divisão inteira do primeiro deles pelo segundo).

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

9. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta verdadeiro, caso o primeiro seja menor do que o segundo, ou falso, caso contrário.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

10. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta verdadeiro, caso o primeiro seja menor ou igual ao segundo, ou falso, caso contrário.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

11. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta verdadeiro, caso o primeiro seja maior do que o segundo, ou falso, caso contrário.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

12. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta verdadeiro, caso o primeiro seja maior ou igual ao segundo, ou falso, caso contrário.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.

13. Faça uma ÚNICA função que, recebendo como parâmetro dois números inteiros, resulta verdadeiro, caso o primeiro seja diferente do segundo, ou falso, caso contrário.

Importante lembrar que os números com os quais vocês trabalharão são inteiros, logo, podem ser positivos ou negativos

Fica proibido declarar variáveis, fazer uso de operadores aritméticos (+, -, \*, / e %), dos operadores relacionais (!=, <, <=, > e >=), de operadores lógicos (!, && e ||), dos comandos de seleção (if com else e switch), de comandos repetitivos (while, do-while e for), try-catch e de funções auxiliares e de qualquer função de biblioteca ou de sua própria autoria, exceto funções implementadas em outros itens desta lista de exercícios.

Dica: use ++, --, ==, if sem else, return, recursão.

Faça um programa para testar sua função.