

# Dredd - Juiz Online

[Principal](#)[Perfil](#)[Minhas Provas](#)[Sair](#)

**Minutos Restantes:**  
433

**Usuário:**  
Luiz Fernando da Silva

**Notas:**  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

## Modularização e Recursão

**Prova Aberta Até:** 15/06/2020 23:00:00

**Número Máximo de Tentativas:** 10

**Atenuação da Nota por Tentativa:** 0%

**Instruções para a prova:** Lista de exercícios sobre modularização para todas as turmas de IAlg. Pode ser acessada de casa.

### Questão 1: Modularização – Soma de dois números

Escreva um programa com uma função que recebe dois parâmetros inteiros, calcula e retorna sua soma. O programa deve pedir ao usuário que digite dois números e então usar esta função para calcular a soma. A saída do resultado deve ser efetuada no programa principal, não na função.

Obs: para soluções em Python deve existir um subprograma chamado **principal**.

#### Entradas:

1. Dois números inteiros (na mesma linha).

#### Saídas:

1. Resultado da soma.

#### Exemplo de Entradas:

2 5

#### Exemplo de Saída:

7

**Peso:** 1

**Última tentativa realizada em:** 12/06/2020 17:48:14

**Tentativas:** 1 de 10

**Nota (0 a 100):** 100

**Status ou Justificativa de Nota:** Nenhum erro encontrado.

[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#)

Nenhum arquivo selecionado

[Enviar Resposta](#)

**Minutos Restantes:**  
433

**Usuário:**  
Luiz Fernando da Silva

**Notas:**

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

## Questão 2: Modularização - Reverso de um número

Faça um programa que receba um dado valor inteiro e retorne o reverso desse valor. Sendo o número reverso formado pela troca de posição entre a ultima casa com a primeira, e penúltima com a segunda e assim por diante.

O reverso precisa ser criado dentro de uma função, mas sua impressão será executada no programa principal.

Obs: soluções em Python devem possuir um subprograma chamado **principal**.

Entradas:

1. Número inteiro que se deseja obter o reverso.

Saídas:

1. Reverso do valor informado pelo usuário.

Exemplo de Entrada:

172

Exemplo de Saída:

271

Exemplo de Entrada:

1459

Exemplo de Saída:

9541

**Peso: 1**

**Última tentativa realizada em:** 12/06/2020 18:01:07**Tentativas:** 1 de 10**Nota (0 a 100):** 100**Status ou Justificativa de Nota:** Nenhum erro encontrado.**Minutos Restantes:**  
433**Usuário:**  
Luiz Fernando da Silva**Notas:**  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#)

Nenhum arquivo selecionado

[Enviar Resposta](#)

### Questão 3: Modularização - Ou-Exclusivo XOR

Ou exclusivo ou disjunção exclusiva, conhecido geralmente por XOR, é uma operação lógica entre dois operandos que resulta em um valor lógico verdadeiro se e somente se exatamente um dos operandos possui valor verdadeiro. A tabela verdade do XOR é (1 é True e 0 é False):

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Crie uma função que receba dois valores lógicos como parâmetro e retorne o resultado da operação XOR. Seu programa receberá do usuário dois valores lógicos da entrada escritos na mesma linha, separados por espaço ("1" ou "0") e deverá testar a função com esses parâmetros. A saída consiste em escrever "True" ou "False".

DICA: No computador, o inteiro "0" representa "False" e "1" representa "True", assim a conversão `bool(1) = True` e `bool(0) = False`.

#### Entrada:

- Dois valores inteiros lógicos (1 ou 0), na mesma linha.

#### Saída:

- Um valor lógico (True ou False).

#### Exemplo de Entrada:

1 0

**Exemplo de Saída:**

True

**Exemplo de Entrada:**

1 1

**Exemplo de Saída:**

False

**Peso:** 1**Última tentativa realizada em:** 12/06/2020 17:49:17**Tentativas:** 1 de 10**Nota (0 a 100):** 100**Status ou Justificativa de Nota:** Nenhum erro encontrado.[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#)

Nenhum arquivo selecionado

[Enviar Resposta](#)**Questão 4: Modularização - Números Harshad**

Na matemática, um número Harshad é um número inteiro que é divisível pela soma dos seus dígitos. O número 81, por exemplo, é um número Harshad pois  $8 + 1 = 9$ , e  $81 \% 9 = 0$ .

Escreva uma função que receba um inteiro, e retorne **um valor lógico** que diz se ele é um número Harshad ou não. O processamento dessa função deverá chamar **outra função** que receberá um inteiro e deverá retornar **a soma de seus dígitos**.

Utilize ambas funções em um programa que terá como entrada um número (inteiro e positivo), e deverá exibir como saída o valor lógico retornado pela função.

**Dica:** transforme o número em uma string para obter a soma dos dígitos.

Obs: soluções em Python devem possuir um subprograma chamado **principal**.

**Minutos Restantes:**  
433

**Usuário:**  
Luiz Fernando da Silva

**Notas:**  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

**Entradas:**

1. Um número inteiro e positivo.

**Saídas:**

1. Um valor lógico que diz se é um número Harshad ou não.

**Exemplo de Entrada:**

81

**Exemplo de Saída:**

True

**Exemplo de Entrada:**

143

**Exemplo de Saída:**

False

**Peso:** 1

**Última tentativa realizada em:** 12/06/2020 18:25:19

**Tentativas:** 3 de 10

**Nota (0 a 100):** 100

**Status ou Justificativa de Nota:** Nenhum erro encontrado.

[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#)

Nenhum arquivo selecionado

[Enviar Resposta](#)

### Questão 5: Modularização - Conversão de unidade

Você foi designado para fazer um programa que converte unidades de distância, que pode ser de metros para centímetros ou vice-versa. Para isso, seu programa **deve ter duas funções**, uma para converter de metros para centímetros e outra para converter o inverso.

**Minutos**  
**Restantes:**  
433

**Usuário:**  
Luiz Fernando da  
Silva

**Notas:**  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

Cada função deve receber a **distância como parâmetro (número real)** e retornar a distância convertida.

**ATENÇÃO!** As funções de conversão devem ser utilizadas na função main, onde serão lidos os valores da distância e da unidade para qual a distância deve ser convertida. Na função main, **depois de utilizada a função correta** deverá ser mostrada na tela a distância convertida.

**OBS:** Caso seja fornecida uma unidade inválida, o programa deverá exibir a palavra **ERRO** como resposta.

Entradas:

1. Distância (real);
2. Unidade para qual deverá ser convertida (minúsculo, singular e sem acento)

Saída:

1. Distância convertida, seguida da abreviação da unidade (m ou cm)

Exemplo de Entrada:

10  
centimetro

Exemplo de Saída:

1000 cm

Exemplo de Entrada 2:

250  
metro

Exemplo de Saída 2:

2.5 m

Exemplo de Entrada 3:

45.7  
VOLT

Exemplo de Saída 3:

ERRO

**Peso: 1**

**Última tentativa realizada em:** 12/06/2020 18:26:20

**Tentativas:** 1 de 10**Nota (0 a 100):** 100**Status ou Justificativa de Nota:** Nenhum erro encontrado.[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#)

Nenhum arquivo selecionado

[Enviar Resposta](#)**Minutos Restantes:**  
433**Usuário:**  
Luiz Fernando da Silva**Notas:**

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

### Questão 6: Modularização - Maior elemento de cinco

Escreva um programa que tem uma função que calcula o maior elemento dentre cinco inteiros. Fazer um projeto dos parâmetros necessários para esta função é parte da avaliação.

O programa deverá ter também um módulo principal que é responsável por toda a entrada e saída de dados do programa (operações de leitura e escrita). Ele lê os elementos a serem processados e escreve todos os elementos novamente, seguidos do maior deles.

#### Entradas:

1. Cinco números inteiros.

#### Saídas:

1. Os cinco números lidos.
2. O maior dos cinco elementos.

#### Exemplo de entradas:

12 3 50 20 2

#### Exemplo de saídas:

12 3 50 20 2  
50

**Peso:** 1**Última tentativa realizada em:** 12/06/2020 18:26:45**Tentativas:** 1 de 10**Nota (0 a 100):** 100

**Status ou Justificativa de Nota:** Nenhum erro encontrado.[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#) Nenhum arquivo selecionado[Enviar Resposta](#)**Minutos Restantes:**  
433**Usuário:**  
Luiz Fernando da Silva**Notas:**

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

### Questão 7: Modularização – Passagem por referência

Escreva um programa que contenha um procedimento que recebe dois parâmetros inteiros, calcula a soma deles e armazene-a em um terceiro parâmetro, que deverá ser passado por referência para que seu valor possa ser alterado dentro do subprograma. O programa deve pedir ao usuário que digite dois números e então usar esse procedimento para calcular a soma e mostrar o resultado na tela.

**Entrada:**

1. Dois valores inteiros, passados como parâmetros para o procedimento.

**Saída:**

1. A soma desses valores, passada por referência pelo procedimento.

**Exemplo de entrada:**

1 2

**Exemplo de saída:**

3

**Peso:** 1**Última tentativa realizada em:** 12/06/2020 18:27:19**Tentativas:** 1 de 10**Nota (0 a 100):** 100**Status ou Justificativa de Nota:** Nenhum erro encontrado.[Ver Código da Última Tentativa](#)



Nova Resposta:

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:

Q1: 100

Q2: 100

Q3: 100

Q4: 100

Q5: 100

Q6: 100

Q7: 100

Q8: 3

Q9: ?

Q10: ?

Q11: ?

Q12: ?

Q13: ?

Q14: ?

Q15: ?

Q16: ?

Q17: ?

Total: 41

### Questão 8: Modularização - Eliminados da Copa

Na primeira fase da Copa do Mundo, as seleções são divididas em grupos. Cada grupo tem 4 seleções e aquelas que tiverem as duas menores pontuações são eliminadas da Copa.

Faça um programa que dada uma lista com os nomes das quatro seleções de um grupo e uma lista com suas respectivas pontuações, exibe quais as duas seleções do grupo que são eliminadas da Copa.

O programa deve ter uma **função que recebe como parâmetro a lista de pontos e retorna as posições das duas menores pontuações**. As posições retornadas pela função devem ser utilizadas no programa principal para exibir os nomes das seleções. Os nomes devem aparecer na ordem da menor para a maior pontuação.

As entradas e saídas do programa devem ser feitas somente no subprograma **principal**.

Assuma que não haverá empate nas pontuações das seleções.

Entradas:

1. Uma lista com os nomes de quatro seleções;
2. Uma lista com as pontuações das respectivas seleções.

Saídas:

1. Os nomes das seleções que são eliminadas da Copa (as que têm as duas menores pontuações).

Exemplo de Entrada:

Peru Australia Franca Dinamarca

3 1 7 5

Exemplo de Saída:

Australia Peru

Exemplo de Entrada:

Brasil Servia Suica Inglaterra

4 2 3 6

Exemplo de Saída:

Servia Suica

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

**Peso: 1****Última tentativa realizada em:** 15/06/2020 03:42:10**Tentativas:** 2 de 10**Nota (0 a 100):** 3

**Status ou Justificativa de Nota:** O programa não resolve todas as instâncias do problema. A quantidade de dados escritos pelo programa é diferente da quantidade de dados esperados.

[Ver Código da Última Tentativa](#)

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

[Escolher arquivo](#)

Nenhum arquivo selecionado

[Enviar Resposta](#)

### Questão 9: Modularização - Sucessor e Antecessor

Escreva um programa que tem uma função que receba um número inteiro e retorne seu antecessor e sucessor. Por exemplo, para o número 2 a função deve retornar 1 e 3. O módulo principal deve chamar a função e imprimir o antecessor, o número passado e o sucessor.

Comandos de leitura ou escrita só podem ser usados no módulo **principal**.

#### Entradas:

1. Número inteiro o qual quer obter o sucessor e o antecessor.

#### Saídas:

1. O antecessor do número.
2. O número.
3. O sucessor do número.

#### Exemplo de entrada:

5

**Exemplo de saída:**

4 5 6

**Minutos**  
**Restantes:**  
433

**Usuário:**  
Luiz Fernando da  
Silva

**Notas:**  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

**Questão 10: Modularização - Divisores de um número**

Escreva um programa que possua um subprograma que recebe um número inteiro e mostre na tela os seus divisores, e também retorne: a) o seu segundo menor divisor; e b) o seu segundo maior divisor.

Por exemplo, para o número 21 a função deve verificar e mostrar os números 1, 3, 7 e 21, que são todos os seus divisores, e também retornar para o programa principal os números 3 e 7, pois o número 3 é o segundo menor divisor de 21, e o número 7 é o segundo maior divisor de 21.

O programa principal deve utilizar a função e mostrar, na tela do monitor, os números do segundo menor divisor e do segundo maior divisor, em uma única linha.

**Observações:**

1. A impressão dos divisores deve ser feita dentro da função;
2. A impressão do segundo menor divisor e do segundo maior divisor deve ser feita no programa principal;
3. Atente para o caso dos números primos, em que o menor divisor é 1 e o maior divisor é o próprio número. Como não há segundo menor divisor e segundo maior divisor, imprima o menor e o maior
4. Para soluções em Python deve existir um subprograma chamado **principal**.

**Entradas:**

1. Um número inteiro

**Saídas:**

1. Todos os divisores
2. O segundo menor divisor
3. O segundo maior divisor

**Minutos Restantes:**  
433

**Usuário:**  
Luiz Fernando da Silva

**Notas:**  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

### Exemplo de entrada:

21

### Exemplo de saída:

1  
3  
7  
21  
3 7

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

## Questão 11: Modularização - Somatório, combinação e fatorial

Faça um programa que receba no usuário um número inteiro positivo e não nulo do usuário ( $n \geq 1$ ) e calcule e exiba o valor da seguinte expressão:

$$\sum_{k=1}^n C_{n,k}, \text{ sendo que } C_{n,k} = \frac{n!}{k!(n-k)!}$$

Com essas expressões em mente, seu programa deverá ser feito utilizando modularização, sendo criada uma função "somatório", que calcula o somatório das combinações da expressão, uma função "combinação", que calcula cada combinação pedida, e uma função "fatorial" para o cálculo de cada fatorial dentro da combinação.

Organize seu programa de forma que cada função dependa uma da outra, ou seja, o cálculo da expressão seja composto por várias chamadas de funções (uma função chamando a outra).

### Entrada:

- Um número inteiro positivo não nulo ( $n \geq 1$ ).

### Saída

- Um número inteiro - o valor do cálculo da expressão (somatório).

**Exemplo de Entrada:**

5

**Exemplo de Saída**

31

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

**Notas:**

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Nenhum arquivo selecionado

**Questão 12: Modularização - Números primos em um intervalo**

Faça um programa que recebe dois números inteiros  $N$  e  $M$ ,  $N \leq M$ , e calcula quantos números primos existem nesse intervalo. O cálculo da quantidade de primos deve ser feito em uma função, que tem como parâmetros os valores de início e fim do intervalo,  $N$  e  $M$ , respectivamente. A leitura e a saída dos dados devem ser realizadas no subprograma principal.

Entradas:

1. Dois valores inteiros  $N$  e  $M$ ,  $N \leq M$  (na mesma linha).

Saídas:

1. A quantidade de números primos no intervalo  $[N, M]$ .

Exemplo de entrada:

3 13

Exemplo de saída:

5

**Peso: 1**

Nova Resposta:

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

 Nenhum arquivo selecionado

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

### Questão 13: Modularização – Somatório com alternância de sinal

$$1^1 - 2^2 + 3^3 - 4^4 + 5^5 - 6^6 + 7^7 - 8^8 + 9^9 - \dots$$

Escreva um programa que calcula uma certa quantidade de termos da série acima. Esta quantidade é uma entrada do programa e deve ser um número inteiro maior que zero. Se o usuário informar um número inteiro menor que 1, o programa deve ler um novo valor, repetidamente, até que o valor seja maior que zero.

O programa deve ter uma função que recebe o número de termos e retorna o somatório dos termos.

As operações de leitura e escrita devem estar na função principal. Para programas em Python, deve ser criada uma função cujo nome é "principal".

Entradas:

1. A quantidade de termos a ser somada (número inteiro maior que zero, sujeito a releitura, conforme explicado acima)

Saídas:

1. O valor do somatório

Exemplo de Entrada 1:

4

Exemplo de Saída 1:

20

Exemplo de Entrada 2:

-5  
25

Exemplo de Saída 2:

-7895

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

**Notas:**

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

**Questão 14: Recursividade - Número áureo**

O *número áureo*, frequentemente denotado pela letra grega  $\phi$  (phi) é um número real irracional que ocorre espontaneamente na natureza e é frequentemente usado nas artes por estar relacionado à nossa percepção de beleza.

O *número áureo* pode ser calculado pela recorrência  $\phi = 1 + 1/\phi$ .

Por ser uma recorrência infinita, ela precisa ser limitada para ser usada na recursividade da Computação. Podemos definir o valor aproximado de  $\phi$  em função do número de termos usados no cálculo, assim:

$$\phi(n) = \begin{cases} 1, & \text{se } n = 1; \\ 1 + \frac{1}{\phi(n-1)}, & \text{se } n > 1. \end{cases}$$

Faça um programa que tem uma função que calcula uma aproximação do *número áureo*, usando recursão.

O *número áureo*, deve ser do tipo *ponto flutuante de precisão dupla* (double) para possibilitar a precisão necessária nos cálculos. As operações de leitura e escrita devem ser realizadas na função principal.

Entradas:

1. O número de termos para o cálculo da aproximação do *número áureo*.

Saídas:

1. O valor aproximado do *número áureo*.

Exemplo de Entrada:

3

Exemplo de Saída:

1.5

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:

Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

**Questão 15: Recursividade – Impressão por recursão**

Crie um subprograma **recursivo** que receba um número inteiro N e imprima todos os números naturais de 0 até N em ordem crescente.

**Entradas:**

1. Um número inteiro

**Saídas:**

1. Uma sequência de números naturais de 0 até N

**Exemplo de Entradas:**

15

**Exemplo de Saída:**

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

**Questão 16: Recursividade - Torre de Hanoi**

O número mínimo  $T(N)$  de movimentos necessários para resolver o problema das Torre de Hanoi com N discos é dado pela recorrência:

- $T(1) = 1$ .
- $T(N) = 2T(N-1)+1$ , para  $N > 1$ .



Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

Faça um programa que tem uma função recursiva para calcular  $T(N)$ . O programa também deve ter uma função principal que faz as leituras, as escritas e executa a função de cálculo. O planejamento da passagem de parâmetros faz parte da avaliação.

#### Entradas:

- O número de discos (N)

#### Saídas

- O número mínimo de movimentos  $T(N)$ .

#### Exemplos de Entradas e Saídas:

##### Entrada:

5

##### Saída:

31

##### Entrada:

8

##### Saída:

255

#### Peso: 1

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta

### Questão 17: Recursividade - Funções mutuamente recursivas

Escreva um programa que calcula o valor da função  $F$ , definida recursivamente:

$F(x) = 1$ , se  $x$  igual a 0,

$F(x) = 2H(x) + F(x-1)$ , se  $x$  maior que 0 e par,

$F(x) = 2H(x) - F(x-1)$ , se  $x$  maior que 0 e ímpar.

Minutos  
Restantes:  
433

Usuário:  
Luiz Fernando da  
Silva

Notas:  
Q1: 100  
Q2: 100  
Q3: 100  
Q4: 100  
Q5: 100  
Q6: 100  
Q7: 100  
Q8: 3  
Q9: ?  
Q10: ?  
Q11: ?  
Q12: ?  
Q13: ?  
Q14: ?  
Q15: ?  
Q16: ?  
Q17: ?  
Total: 41

Observe que  $F$  é definida em termos de  $H$ , apresentada abaixo:

$H(x) = 0$ , se  $x$  igual a 0,  
 $H(x) = H(x-1) + F(x-1)$ , se  $x$  maior que 0.

### Entradas:

1. Valor inteiro  $x$  positivo para o qual se deseja calcular o valor de  $F(x)$ .

### Saídas:

1. Valor de  $F(x)$

Exemplo de Entrada:  
8

Exemplo de Saída:  
2705

=====  
Exemplo de Entrada:  
5

Exemplo de saída:  
73

**Peso: 1**

Nova Resposta: \_\_\_\_\_

Selecione o arquivo com o código fonte do programa que resolve o problema para enviá-lo.

Escolher arquivo

Nenhum arquivo selecionado

Enviar Resposta



Desenvolvido por Bruno  
Schneider a partir do programa  
original (Algod) de Renato R.  
R. de Oliveira.

