

Dredd - Juiz Online

Principal

Perfil

Minhas Provas

Sair

Minutos

Restantes:
21647

Usuário:

Lucas Antonio
Lopes Neves

Notas:

Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

IAIg - Exercícios de Modularização

Prova Aberta Até: 16/12/2017 11:00:00

Número Máximo de Tentativas: 6

Atenuação da Nota por Tentativa: 0%

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

Questão 1: Modularização - Capicuas

Um número é dito ser capicua quando lido da esquerda para a direita é o mesmo que quando lido da direita para a esquerda. O ano 2002, por exemplo, é capicua. Implemente uma função que receba como parâmetro **um inteiro** e retorne um valor lógico que indica se o número tem essa característica.

Uma outra função ou procedimento deve receber todos os números que são capicuas e devolver ao módulo principal o maior e o menor deles.

As funções serão parte de um programa cujo módulo principal lê a quantidade de números a serem analisados e escreve os números que não são capicuas, e dentre os capicuas, o menor e o maior valores.

Entrada do programa:

1. Número inteiro representando a quantidade de valores a serem testados.
2. Números inteiros a serem testados.

Saída do programa:

1. Números que não são capicuas
2. Menor número capicua lido
3. Maior número capicua lido

Exemplo de entrada do programa:

```
7
12
1122
131
13412
344312
1111111
4578754
```

Exemplo de saída do programa:

```
12
1122
13412
344312
131
4578754
```

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta

Questão 2: Modularização – Sequência de Fibonacci

Crie um algoritmo que imprima os **n** primeiros termos da sequência de Fibonacci, utilizando uma função que retorna o **n**-ésimo termo da referida série. Os termos dessa série são definidos da seguinte forma:

$Fib(0) = 0$

$Fib(1) = 1$

$Fib(n) = Fib(n-1) + Fib(n-2)$, para **n** maior ou igual à 2

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

Entradas:

1. Um número inteiro (n).

Saídas:

1. Sequência de Fibonacci começando do 0.

Exemplo de Entradas:

10

Exemplo de Saída:

0 1 1 2 3 5 8 13 21 34

Peso: 1

Última tentativa realizada em: 28/11/2017 11:26:35

Tentativas: 3 de 6

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.

Choose File No file chosen

Enviar Resposta

Questão 3: Modularizacao - Palindromos

Um palíndromo é um texto que pode ser lido tanto da esquerda para a direita como da direita para a esquerda. Por exemplo, sem considerar sinais gráficos, acentos e letras maiúsculas e minúsculas: "Socorram-me, subi no ônibus em Marrocos".

Implemente uma função que receba como parâmetro **uma palavra** e retorne um valor lógico que indica se a palavra tem essa característica. Uma outra função deve receber todas as palavras que são palíndromos e devolver ao módulo principal a maior e a menor delas.

As funções serão parte de um programa cujo módulo principal lê a quantidade de palavras a serem analisadas e escreve aquelas que não são palíndromos, e dentre os palíndromos, a menor e a maior delas (a menor é a primeira das palavras quando ordenadas em ordem alfabética, e a maior é a última).

Entradas:

1. Número inteiro representando a quantidade de palavras a serem testadas.
2. Palavras a serem testadas (uma em cada linha).

Saídas:

1. Palavras que não são palíndromos
2. Menor palavra palíndromo lida
3. Maior palavra palíndromo lida

Exemplo de entrada:

```
7
casa
ovo
pelo
tomate
arara
papa
ama
```

Exemplo de saída:

```
casa
pelo
tomate
papa
ama
ovo
```

Peso: 1

Nova Resposta: _____

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

 No file chosen**Questão 4: Modularização - Imposto****Minutos**

Restantes:
21647

Usuário:

Lucas Antonio
Lopes Neves

Notas:

Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Faça um programa para calcular o valor de imposto pago por uma empresa em uma nota fiscal. Para isso o programa deve ser capaz de receber as alíquotas de cada tipo de imposto e os dados da notas fiscais emitidas pela empresa.

Além disso, o programa deve ser organizado como pedido abaixo:

- Os impostos serão armazenados em um vetor de registros contendo o nome do imposto e o valor da alíquota (Ex: ICMS, 0.25).
- A nota fiscal será armazenada em um vetor de registros de itens de nota fiscal, que contém: descrição, valor e nome do imposto.
- Cada item abaixo deve ser feito em um subprograma específico
 - A leitura de cada registro de alíquota de imposto.
 - A leitura de cada item da nota fiscal.
 - A leitura da nota fiscal.
 - O cálculo do imposto pago para cada item da nota.
 - O cálculo do total de imposto pago na nota.

Exemplo de Entrada:

```
3                                     → Quantidade de alíquotas
ICMS 0.25 IPI 0.11 IIS 0.06
2                                     → Quantidade de itens da nota fiscal
Carro 40000 IPI
Computador 3000 IIS
```

Exemplo de Saída:

4580

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta

Questão 5: Modularização - Busca Binária

Implemente um programa que tem uma função para realizar a busca binária em um vetor ordenado de números inteiros.

Toda entrada e saída de dados (leituras e escritas) deve ficar a cargo do módulo principal, enquanto que o processamento propriamente dito (a busca binária) deve ficar numa função específica. Um bom planejamento dos parâmetros da função de busca é importante para uma boa nota.

A função principal deverá ler os dados de entrada, passar as informações para a função e escrever os resultados calculados pela função. Não é permitido fazer a busca duas vezes para obter os dois valores da resposta.

Quando a quantidade de elementos for par, a escolha do meio deve favorecer o elemento da esquerda.

Entradas do programa:

1. Tamanho do vetor.

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

2. Os elementos (números inteiros) do vetor ordenado (já em ordem crescente).
3. Elemento a ser buscado.

Saídas do programa:

1. Posição do elemento no vetor (escreva -1 caso o elemento não tenha sido encontrado).
2. Número de comparações (quantos elementos do vetor foram comparados ao elemento buscado).

Exemplos de entradas e saídas do programa:

Entradas:

```
6
1 4 6 7 8 9
4
```

Saídas:

```
1
3
```

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta

Questão 6: Modularização - Escrever vetor

Faça um programa que lê vários números inteiros, guarda-os num vetor e escreve o conteúdo do vetor.

Deverá existir um subprograma que recebe um vetor via parâmetros e escreve o seu conteúdo. O vetor pode ter tamanho zero. A escrita do vetor deve ser formatada, com os seguintes itens:

- Deve haver um "[" (abre colchetes) antes do primeiro elemento. Não deve haver qualquer espaço entre o "[" e o primeiro elemento.
- Deve haver uma vírgula seguida de espaço entre cada elemento.
- Deve haver um "]" (fecha colchetes) depois do último elemento. Não deve haver qualquer espaço entre o último elemento e o "]".

O subprograma principal deverá escrever um final de linha depois de escrever o conteúdo do vetor.

Obs.: Vetores em Python são implementados como listas cujos elementos são todos homogêneos.

Entradas:

1. A quantidade de números a serem lidos.
2. Vários números inteiros.

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Saídas:

1. O conteúdo do vetor criado, conforme a formatação acima.

Exemplo de Entrada:

```
4
2 3 4 1
```

Exemplo de Saída:

```
[2, 3, 4, 1]
```

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta

Questão 7: Modularização - Inserir em ordem

Faça um programa que cadastra nomes e telefones de pessoas. Para simplificar o desenvolvimento, considere que os nomes não possuem espaços ou seja: "Carlos" é um nome válido para este problema, mas "Carlos Silva" não é. Considere também que telefones são números inteiros de 8 dígitos.

O programa deve manter os nomes cadastrados sempre em ordem alfabética. Para tanto, ele deve ter uma sub-rotina que insere um registro num vetor de registros, mantendo a ordenação no vetor. Um bom planejamento dos parâmetros dessa sub-rotina é parte da avaliação.

Não deve ser implementada uma função de ordenação neste programa. Manter os dados ordenados por repetidas ordenações toda vez que um novo elemento aparece, é extremamente lento e não é uma solução aceitável.

Para comparação de nomes usando vetores de char, é permitido usar a função strcmp da biblioteca cstring. Essa função recebe apenas duas strings e retorna um número inteiro que é negativo se a primeira for menor que a segunda; zero se as strings forem iguais ou um número positivo se a segunda for menor.

Toda entrada e saída de dados (operações de leitura e escrita) deve ser feita no módulo principal. Ele deve ler vários pares (nome, telefone) e, para cada par lido, escrever todos os registros que constam na memória.

Entradas do programa:

1. o número de registros que serão lidos.
2. diversos pares de nomes e telefones.

Saídas do programa:

1. O conjunto completo de registros, cada vez que um novo registro for lido.

Exemplo de entrada:

```
3
Cristina 38210000
```

Carlos 38221234
Maria 38212121

Exemplo de saída:

Cristina 38210000

Carlos 38221234
Cristina 38210000

Carlos 38221234
Cristina 38210000
Maria 38212121

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta

Questão 8: Modularização - Mediana de um vetor

Escreva um programa que recebe vários números inteiros, em ordem crescente e então determina sua mediana.

Deverá existir um subprograma que recebe um vetor ordenado e retorna a mediana.

Todas as operações de leitura e escrita devem ser realizadas no módulo principal.

A mediana de uma sequência ordenada de N elementos (e_0, e_1, \dots, e_{n-1}) é definida da seguinte forma:

1. Se N for ímpar, então a mediana é dada pelo elemento da posição central.
2. Se N for par, então a mediana é dada pela média aritmética dos dois elementos nas posições centrais.

Obs.: Vetores em Python são implementados como listas cujos elementos são todos homogêneos.

Entradas:

1. Número de elementos no vetor (maior que 0).
2. Elementos do vetor. Os elementos têm tipo inteiro.

Saídas:

1. Mediana.

Exemplo de entrada 1:

5
1 3 9 12 20

Exemplo de saída 1:

9

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Exemplo de entrada 2:

4
1 4 7 9

Exemplo de saída 2:

5.5

Peso: 1

Nova Resposta: _____

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

Choose File

No file chosen

Enviar Resposta

Questão 9: Modularização - Filtro de vetor

Faça um programa que tenha um subprograma (função) que receba um vetor de números inteiros e que retorne um novo vetor com os valores do primeiro que são divisíveis por 2 e por 3. Esse subprograma não pode ter nenhuma operação de leitura nem de escrita. Ele também não pode alterar o vetor com os valores de entrada. Um bom planejamento dos parâmetros é fundamental e será parte importante da avaliação.

O programa principal deverá ler os valores e escrever todos os números que passaram pelo filtro da função.

OBS.: A verificação de divisibilidade dos números só pode ser feita dentro da função.

OBS2: Vetores em Python são implementados como listas cujos elementos são todos homogêneos.

Entradas:

1. A quantidade de números.
2. Os números inteiros que devem passar pelo filtro da função.

Saídas:

1. Todos os números que passaram pelo filtro, na mesma ordem em que foram lidos.

Exemplo de entrada:

7
36 37 0 41 7 12 -9

Exemplo de saída:

36 0 12

Peso: 1

Nova Resposta:

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

Choose File

No file chosen

Enviar Resposta

Minutos

Restantes:

21647

Usuário:

Lucas Antonio

Lopes Neves

Notas:

Q1: ?

Q2: 100

Q3: ?

Q4: ?

Q5: ?

Q6: ?

Q7: ?

Q8: ?

Q9: ?

Q10: ?

Q11: ?

Q12: ?

Q13: ?

Q14: ?

Total: 7

Questão 10: Modularização - Fogueira de Ingaí

A cidade de Ingaí, próxima a Lavras, realiza todos os anos a Festa de São João na qual a grande atração é a tradicional fogueira. Isso porque não é uma fogueira qualquer, mas sim uma das mais altas do Brasil. Antigamente a cada ano a fogueira era mais alta que a do ano anterior. Mas ultimamente, por questões de segurança, a altura da fogueira tem variado a cada ano (não sendo necessariamente maior que a do ano anterior).

Para registrar a altura da fogueira a cada festa, alguém anotou a partir de um determinado ano **A**, qual era a altura da fogueira naquele ano, e depois anotou em cada ano o quanto a fogueira foi mais alta ou mais baixa que no anterior. Faça então um programa que receba como entrada o ano **A**, a altura da fogueira no ano **A**, o tamanho do vetor e o vetor de inteiros contendo a diferença de altura da fogueira a cada ano. Construa também um **subprograma** que receba os dados necessários por parâmetro, e retorne, **por referência**, os dois anos nos quais a fogueira foi mais alta. Por fim, exiba os dois anos retornados (primeiro o que teve a fogueira mais alta e depois o que teve a segunda fogueira mais alta).

Obs: toda operação de leitura e escrita deve ser feita no subprograma principal; considere que não existem alturas mais altas repetidas.

Obs.2: Em Python os vetores são representados como listas

Obs. 3: Passegem por referência só se aplica a exercícios resolvidos em C++

Entradas:

1. Ano inicial **A** quando foi feita a primeira medição da altura da fogueira.
2. Altura da fogueira no ano **A** (número inteiro).
3. Tamanho do vetor de inteiros com as diferenças de altura.
4. Vetor de inteiros com as diferenças de altura da fogueira a partir do ano **A**. Ou seja, o primeiro elemento do vetor tem a diferença da altura da fogueira do ano **A+1** para o ano **A**; o segundo elemento a diferença de altura do ano **A+2** para o ano **A+1**, e assim sucessivamente.

Saídas:

1. Ano que teve a fogueira mais alta.
2. Ano que teve a fogueira com a segunda maior altura.

Exemplo de Entrada:

```
2000
30
6
3 -5 4 2 -2 -1
```

Nesse exemplo, a altura da fogueira em 2000 foi 30m, em 2001 foi 33m, em 2002 foi 28m, em 2003 foi 32m, e assim sucessivamente.

Exemplo de Saída:

```
2004
2001
```

Exemplo de Entrada:

2011
40
5
-10 8 -6 4 -2

Exemplo de Saída:

2011
2013

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Peso: 1

Nova Resposta: _____

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

 No file chosen**Questão 11: Modularização - Subsequências decrescentes**

Faça uma função que receba como parâmetros um vetor de números inteiros e o tamanho deste vetor. Assuma que todos os valores armazenados no vetor são diferentes entre si. Sua função deve retornar a quantidade de subsequências decrescentes presentes no vetor. Por exemplo, assumindo um vetor com os valores {6, 5, 1, 2, 3, 9, 8, 4, 7, 10} é possível identificar as subsequências decrescentes {6, 5, 1} e {9, 8, 4}. Sendo assim, a função a ser desenvolvida deve retornar o valor 2. A função deve ser implementada de modo que nenhuma operação de entrada e saída de dados seja realizada na mesma.

Obs.: Vetores em Python são implementados como listas cujos elementos são todos homogêneos.

Entrada:

1. Número inteiro denotando o tamanho do vetor;
2. Sequência de números inteiros denotando os valores a serem armazenados no vetor.

Saída:

1. Número inteiro denotando a quantidade de subsequências decrescentes no vetor.

Exemplo de entrada:

10
6 5 1 2 3 9 8 4 7 10

Exemplo de saída:

2

Peso: 1

Nova Resposta: _____

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

No file chosen

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Questão 12: Modularização - Dígitos de um número

Faça um programa que tem um subprograma que recebe um número inteiro não negativo e retorna um vetor com todos os seus dígitos. Esse subprograma não pode ter nenhuma operação de leitura nem de escrita. Os números podem ser retornados na ordem direta ou inversa pelo subprograma. Um bom planejamento dos parâmetros é fundamental e será parte importante da avaliação.

É permitido fazer outros subprogramas para auxiliar a criação deste.

O subprograma principal deverá ler o número inteiro e escrever todos os dígitos encontradas pelo outro subprograma.

Obs.: Vetores em Python são implementados como listas cujos elementos são todos homogêneos.

Entradas:

1. Um número inteiro.

Saídas:

Todos os dígitos do número da entrada na mesma ordem, separados com espaços.

Exemplo de entrada:

812

Exemplo de saída:

8 1 2

Peso: 1

Nova Resposta: _____

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

 No file chosen

Questão 13: Modularização - Série de Fetuccine.

Crie um algoritmo que imprima os n primeiros termos da sequência de Fetuccine, utilizando uma função que retorna o n -ésimo termo da referida série. Nesta série, os dois primeiros termos são fornecidos pelos usuários e os demais obedecem à seguinte regra de recorrência:

$$F(n) = F(n-1) + F(n-2), \text{ para } n \geq 2, \text{ se } n \text{ é ímpar};$$

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

$F(n) = F(n-1) - F(n-2)$, para $n \geq 2$, se n é par.

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

Entradas:

1. Número de termos da sequência(int).
2. Dois primeiros termos da sequência(int).

Saídas:

1. N primeiros termos da sequência de Fetuccine.

Exemplo de Entrada:

10
1 2

Exemplo de Saída:

1 2 3 1 4 3 7 4 11 7

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta

Questão 14: 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

Minutos
Restantes:
21647

Usuário:
Lucas Antonio
Lopes Neves

Notas:
Q1: ?
Q2: 100
Q3: ?
Q4: ?
Q5: ?
Q6: ?
Q7: ?
Q8: ?
Q9: ?
Q10: ?
Q11: ?
Q12: ?
Q13: ?
Q14: ?
Total: 7

Peso: 1

Nova Resposta: _____

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

Choose File No file chosen

Enviar Resposta



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

