

## BCC701 – Programação de Computadores I – 2018-02

### Lista de Exercícios 03: Estruturas de Iteração

<http://www.decom.ufop.br/bcc701/>

#### Questão 1. (2014-01)

Considere o somatório com  $n$  termos definido a seguir:

$$\frac{x+y}{1!} + \frac{3x-7y}{2!} + \frac{5x+13y}{3!} + \frac{7x-19y}{4!} + \frac{9x+25y}{5!} - \dots$$

Escreva um programa Scilab que solicite ao usuário o valor de  $n$ , calcule e imprima o valor desse somatório, considerando que  $x = 50$  e  $y = -80$ . Seu programa deve verificar se o valor de  $n$  digitado pelo usuário é um número positivo,  $n > 0$ , solicitando repetidamente um novo valor, caso o valor digitado não seja válido. Você pode supor que o valor digitado pelo usuário será sempre um número inteiro (você apenas precisa testar se esse valor é positivo).

A seguir, dois exemplos de execução do programa.

#### Exemplo 1:

```
INFORME O NÚMERO DE PARCELAS: 0
VALOR INVÁLIDO PARA n !
INFORME O NÚMERO DE PARCELAS: -6
VALOR INVÁLIDO PARA n !
INFORME O NÚMERO DE PARCELAS: 1

VALOR DO SOMATÓRIO COM 1 PARCELAS: -30
```

#### Exemplo 2:

```
INFORME O NÚMERO DE PARCELAS: 4

VALOR DO SOMATÓRIO COM 4 PARCELAS: 271.25
```

#### Questão 2. (2014-01)

Em uma tubulação projetada para escoamento de água, deve-se avaliar uma grandeza denominada “perda de carga”, a qual depende da vazão de líquido, do diâmetro e da rugosidade do tubo. Utiliza-se para isso a fórmula de Hazen-Williams:

$$J = Q^{1,85} \times 10,64 \times D^{-4,87} \times C^{-1,85}$$

onde:

- $J$  = perda de carga (m/m);
- $Q$  = vazão ( $m^3/s$ );
- $D$  = diâmetro do tubo (m);
- $C$  = coeficiente de rugosidade.

Escreva um programa que leia a vazão desejada em um projeto de tubulação e a perda de carga máxima permitida. A seguir, o programa deve ler o diâmetro e o coeficiente de rugosidade de vários tubos, até encontrar um que proporcione perda de carga abaixo do limite permitido. Não é necessário verificar a validade dos dados fornecidos pelo usuário. A seguir, um exemplo de execução do programa.

**Exemplo 1:**

Informe a vazão desejada (m<sup>3</sup>/s): 0.1

Informe a perda de carga máxima permitida (m/m): 0.4

Informe o diâmetro do tubo (m): 1

Informe o coeficiente de rugosidade do tubo: 0.1

O tubo não atende as especificações! Tente novamente.

Informe o diâmetro do tubo (m): 1.05

Informe o coeficiente de rugosidade do tubo: 0.05

O tubo não atende as especificações! Tente novamente.

Informe o diâmetro do tubo (m): 2

Informe o coeficiente de rugosidade do tubo: 0.1

O tubo com diâmetro 2 m e coeficiente de rugosidade 0.1

atende as especificações com perda de carga de 0.363853 m/m

**Questão 3.** (2014-01)

Escreva um programa para ler diversos números inteiros positivos e, para cada um deles, determinar e imprimir seus divisores. O processamento deverá terminar imprimindo a frase “FIM DO PROGRAMA”, quando o número inteiro digitado for o número -1. Não é necessário verificar se o número digitado é inteiro e positivo. A seguir, um exemplo de execução do programa.

**Exemplo 1:**

Digite um número inteiro positivo (-1 para terminar): 15

Divisores de 15: 1 3 5 15

Digite um número inteiro positivo (-1 para terminar): 36

Divisores de 36: 1 2 3 4 6 9 12 18 36

Digite um número inteiro positivo (-1 para terminar): -1

**FIM DO PROGRAMA**

**Questão 4.** (2014-02)

Considere o somatório com  $n$  termos definido a seguir:

$$\frac{k+m}{1!} + \frac{2k-2m}{3!} + \frac{3k+3m}{5!} + \frac{4k-4m}{7!} + \frac{5k+5m}{9!} + \dots$$

Escreva um programa Scilab que solicite ao usuário o valor de  $n$ , calcule e imprima o valor desse somatório, considerando os valores constantes para  $k = 30$  e  $m = 40$ .

Suponha que o valor de  $n$  digitado pelo usuário seja um valor inteiro e positivo, ou seja,  $n > 0$ . Não será necessária a validação desta entrada de dado.

A seguir, dois exemplos de execução do programa.

**Exemplo 1:**

**INFORME O NÚMERO DE PARCELAS:** 1

**VALOR DO SOMATÓRIO COM 1 PARCELAS:** 70

**Exemplo 2:**

**INFORME O NÚMERO DE PARCELAS:** 4

**VALOR DO SOMATÓRIO COM 4 PARCELAS:** 68.4087

### Questão 5. (2014-02)

Escreva um programa para controlar uma compra no supermercado, tal como mostrado no exemplo a seguir. O programa inicialmente lê o valor que o cliente tem disponível para realizar sua compra. Em seguida, o programa deve ler o preço de cada produto selecionado pelo cliente, somando esses preços para calcular o valor da compra. Para cada preço de produto lido, deve ser impressa uma das mensagens: “**Produto incluído**”, se a soma do preço do produto ao valor da compra não resulta em valor maior do que o disponível pelo cliente; ou “**Produto não incluído**”, em caso contrário, sendo que, neste caso, o preço do produto não deve ser adicionado ao valor da compra. Para cada produto lido, o programa deve também imprimir o valor ainda disponível pelo cliente.

A compra se encerra quando o valor digitado para um produto for igual a **0**. Quando isso ocorrer, o programa deve imprimir o valor disponível inicialmente, o valor total da compra e a mensagem “**FIM DO PROGRAMA**”,

Não é necessária a validação dos dados de entrada; supõe-se que sempre serão digitados valores corretos.

#### Exemplo 1:

```
Controle de uma compra no supermercado
Digite o valor disponível: 400
Digite o preço de um produto (ou zero para encerrar): 80

>>> Produto incluído
Valor ainda disponível: 320.00
Preço de outro produto (ou zero para encerrar): 58

>>> Produto incluído
Valor ainda disponível: 262.00
Preço de outro produto (ou zero para encerrar): 280

>>> Produto não incluído
Valor ainda disponível: 262.00
Preço de outro produto (ou zero para encerrar): 290

>>> Produto não incluído
Valor ainda disponível: 262.00
Preço de outro produto (ou zero para encerrar): 38

>>> Produto incluído
Valor ainda disponível: 224.00
Preço de outro produto (ou zero para encerrar): 49

>>> Produto incluído
Valor ainda disponível: 175.00
Preço de outro produto (ou zero para encerrar): 0

Valor disponível inicialmente: 400.00
Total da compra: 225.00
FIM DO PROGRAMA
```

**Questão 6.** (2014-02)

Bart Simpson está aprendendo a jogar xadrez e tem dificuldade em saber para qual direção ele pode mover sua Torre. Sabemos que um tabuleiro de xadrez é composto por 8 linhas e 8 colunas, e que a Torre se move ortogonalmente, ou seja, pelas linhas (horizontais) e pelas colunas (verticais). Escreva um programa em Scilab que solicite ao Bart as entradas do número da linha e da coluna que indicam a posição de sua Torre. O programa imprime quais são os possíveis movimentos da Torre. Utilize o valor **0** para indicar uma casa para a qual a Torre não pode ser movida e o valor **1** para indicar uma casa para a qual ela pode ser movida. Veja o exemplo de execução abaixo.

**Exemplo 1:**

```
Movimentos de uma Torre no xadrez.
Digite a linha em que a torre se encontra: 6
Digite a coluna em que a torre se encontra: 4
```

**Movimentos possíveis:**

	1	2	3	4	5	6	7	8
1	0	0	0	1	0	0	0	0
2	0	0	0	1	0	0	0	0
3	0	0	0	1	0	0	0	0
4	0	0	0	1	0	0	0	0
5	0	0	0	1	0	0	0	0
6	1	1	1	1	1	1	1	1
7	0	0	0	1	0	0	0	0
8	0	0	0	1	0	0	0	0

**Questão 7.** (2015-01)

O valor de  $\pi$  pode ser aproximado pelo somatório infinito:

$$\pi \approx 3 + \frac{4}{2 \times 3 \times 4} - \frac{4}{4 \times 5 \times 6} + \frac{4}{6 \times 7 \times 8} - \frac{4}{8 \times 9 \times 10} + \frac{4}{10 \times 11 \times 12} - \dots$$

Escreva um programa para calcular um valor aproximado para  $\pi$  de acordo com a fórmula acima. O programa deve ler o número **n** de parcelas a serem usadas no cálculo do somatório. O programa deve testar se valor de **n** é um inteiro positivo, solicitando que seja digitado novo valor de **n**, repetidamente, enquanto o valor lido não for válido.

**Exemplo 1:**

```
QUAL A QUANTIDADE DE PARCELAS (n)?: 0
ERRO: VALOR INVÁLIDO PARA n.
QUAL A QUANTIDADE DE PARCELAS (n)?: -1
ERRO: VALOR INVÁLIDO PARA n.
QUAL A QUANTIDADE DE PARCELAS (n)?: 2.2
ERRO: VALOR INVÁLIDO PARA n.
QUAL A QUANTIDADE DE PARCELAS (n)?: 1

VALOR APROXIMADO DE Pi = 3
```

**Exemplo 2:**

```
QUAL A QUANTIDADE DE PARCELAS (n)?: 3

VALOR APROXIMADO DE Pi = 3.13333
```

**Exemplo 3:**

```
QUAL A QUANTIDADE DE PARCELAS (n)?: 1000

VALOR APROXIMADO DE Pi = 3.14159
```

### Questão 8. (2015-01)

Faça um programa que leia o número de alunos de uma turma ( $n$ ) e, em seguida, leia as notas de cada um dos alunos e determine e imprima os seguintes dados:

- A média das notas dos alunos
- O número de alunos aprovados, isto é, com nota igual ou maior que 6,0.

**OBS:** O programa não precisa verificar se o valor lido para o número de alunos ou para cada uma das notas é válido, podendo supor que sempre será digitado um valor válido.

Um exemplo de execução é mostrado a seguir.

#### Exemplo 1:

SISTEMA DE NOTAS

NÚMERO DE ALUNOS DA TURMA: 7

DIGITE A NOTA DO ALUNO 1: 8.5  
DIGITE A NOTA DO ALUNO 2: 4.5  
DIGITE A NOTA DO ALUNO 3: 7.3  
DIGITE A NOTA DO ALUNO 4: 9.2  
DIGITE A NOTA DO ALUNO 5: 6.4  
DIGITE A NOTA DO ALUNO 6: 3.7  
DIGITE A NOTA DO ALUNO 7: 7.8

MÉDIA DAS NOTAS = 6.77

NÚMERO DE ALUNOS APROVADOS = 5

### Questão 9. (2015-01)

O setor de compras do UFOP está realizando uma cotação de preços para comprar 10 novos computadores para o laboratório de computação. Você vai escrever um programa para auxiliar a UFOP nessa tarefa. O programa deve ler os seguintes dados:

1. O valor que a UFOP tem disponível para a compra de novos computadores.
2. O preço do computador, fornecido por cada uma das empresas nas quais foi feita a cotação de preços. A leitura desses valores deve terminar quando for digitado um valor menor ou igual a 0 (zero) para o preço do computador.

O programa deve imprimir os seguintes dados:

1. O preço mínimo, e o número da empresa que ofereceu o preço mínimo.
2. Caso o dinheiro disponível seja suficiente para comprar os 10 novos computadores (pelo preço mínimo), o programa deve informar qual será o saldo restante. Caso contrário deverá informar qual é o maior número de computadores que é possível adquirir com o dinheiro disponível e qual é o saldo restante.

**OBS:** Não é necessário validar os dados de entrada, isto é, supõe-se que sempre serão digitados valores válidos. A entrada e saída de dados deve ser conforme mostrado nos exemplos de execução a seguir.

**Exemplo 1:**

Compra de Computadores

Digite o valor disponível: 15000.00

Preço do produto na loja 1(ou zero para encerrar): 2300.00

Preço do produto na loja 2(ou zero para encerrar): 1950.00

Preço do produto na loja 3(ou zero para encerrar): 1480.00

Preço do produto na loja 4(ou zero para encerrar): 990.00

Preço do produto na loja 6(ou zero para encerrar): 0

Menor preço = R\$ 990.00 (empresa 4)

Os computadores podem ser comprados

O saldo restante será de R\$ 5100.00

**Exemplo 2:**

Compra de Computadores

Digite o valor disponível: 7000.00

Preço do computador na loja 1(ou zero para encerrar): 2300.00

Preço do computador na loja 2(ou zero para encerrar): 1950.00

Preço do computador na loja 3(ou zero para encerrar): 1480.00

Preço do computador na loja 4(ou zero para encerrar): 990.00

Preço do computador na loja 6(ou zero para encerrar): 0

Menor preço = R\$ 990.00 (loja 4)

Só podem ser comprados 7 computadores

O saldo restante será de R\$ 70.00

**Questão 10. (2015-02)**

O valor de  $e^x$  pode ser aproximado pelo somatório infinito:  $e^x \approx 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \frac{x^5}{5!} + \dots$

Escreva um programa para calcular um valor aproximado para  $e^x$  de acordo com a fórmula acima. O programa deve ler o valor de  $x$  e o número  $n$  de parcelas a serem usadas no somatório. O programa deve testar se valor de  $n$  é um inteiro positivo, solicitando que seja digitado novo valor de  $n$ , repetidamente, enquanto o valor lido não for válido. Três exemplos de execução do programa são apresentados a seguir.

**Exemplo 1:**

Digite o valor de x: 2.3

Digite o número de parcelas: 0

ERRO: VALOR INVÁLIDO.

Digite o número de parcelas: -1

ERRO: VALOR INVÁLIDO.

Digite o número de parcelas: 2.2

ERRO: VALOR INVÁLIDO.

Digite o número de parcelas: 1

**VALOR APROXIMADO DE  $e^{2.3}$  COM 1 PARCELA = 1**

**Exemplo 2:**

Digite o valor de x: 2.3

Digite o número de parcelas: 10

**VALOR APROXIMADO DE  $e^{2.3}$  COM 10 PARCELAS = 9.97275**

**Exemplo 3:**

Digite o valor de x: 2.3

Digite o número de parcelas: 20

**VALOR APROXIMADO DE  $e^{2.3}$  COM 20 PARCELAS = 9.97418**

### Questão 11. (2015-02)

O controle de qualidade de uma fábrica de cimento envolve diversos tipos de teste, entre eles, um teste de compressão em amostras de argamassa preparada com o cimento. Você vai fazer um programa para auxiliar nesse controle de qualidade. O programa deve ler, inicialmente, o valor padrão para a carga de compressão das amostras **crp** (em kg/cm<sup>2</sup>) e o número **n** de amostras a serem testadas. Em seguida, para cada amostra, o programa deve ler a sua área **a** (em cm<sup>2</sup>) e o peso **p** (em kg) que provocou a ruptura da amostra, calculando a sua carga de compressão como **p/a**. O programa deve determinar e imprimir o valor mínimo dentre as cargas de ruptura das amostras testadas. Se o valor mínimo dentre as cargas de ruptura das amostras for maior ou igual ao valor padrão (**crp**), o programa deve imprimir uma mensagem informando que o cimento foi aprovado; caso contrário, que o cimento foi reprovado.

**OBS:** O programa não precisa verificar se os valores lidos são válidos, podendo supor que sempre será digitado um valor válido.

Dois exemplos de execução são mostrados a seguir.

#### Exemplo 1:

```
CONTROLE DE QUALIDADE - COMPRESSÃO
Valor padrão da carga de compressão (kg/cm2): 100
Número de amostras: 3

Área da amostra 1 (cm2): 1
Peso de da amostra 1 (kg): 120
Área da amostra 2 (cm2): 1.2
Peso de da amostra 2 (kg): 150
Área da amostra 3 (cm2): 0.9
Peso de da amostra 3 (kg): 110

CARGA DE RUPTURA MÍNIMA = 120
CIMENTO APROVADO.
```

#### Exemplo 2:

```
CONTROLE DE QUALIDADE - COMPRESSÃO
Valor padrão da carga de compressão (kg/cm2): 100
Número de amostras: 3

Área da amostra 1 (cm2): 1
Peso de da amostra 1 (kg): 120
Área da amostra 2 (cm2): 1.5
Peso de da amostra 2 (kg): 110
Área da amostra 3 (cm2): 1.3
Peso de da amostra 3 (kg): 100

CARGA DE RUPTURA MÍNIMA = 73.3333
CIMENTO REPROVADO.
```

### Questão 12. (2015-02)

A produção atual de minério de ferro de duas mineradoras concorrentes é:

- mineradora **A** – 500 toneladas
- mineradora **B** – 700 toneladas

A mineradora **B** desenvolveu um projeto de expansão, prevendo aumento de sua produção a uma taxa de 2% ao ano. Escreva um programa que leia a taxa de aumento de produção prevista pelo plano de expansão da mineradora **A** e determine se a produção de **A** eventualmente será *maior* do que a produção de **B** e, em caso afirmativo, quantos anos serão necessários para que isso aconteça.

A saída do programa deve ser tal como mostrado nos dois exemplos a seguir.

**Exemplo 1:**

**CONCORRÊNCIA**

Taxa de aumento da produção de A: 1.8

MINERADORA A: Produção atual = 500 ton; Taxa de crescimento = 1.8%

MINERADORA B: Produção atual = 700 ton; Taxa de crescimento = 2.0%

A produção de A nunca será maior que a de B.

**Exemplo 2:**

**CONCORRÊNCIA**

Taxa de aumento da produção de A: 3.5

MINERADORA A: Produção = 500 ton; Taxa de crescimento = 3.5%

MINERADORA B: Produção = 700 ton; Taxa de crescimento = 2.0%

A produção de A será maior que a de B em 24 anos.

**Questão 13. (2016-01)**

O valor de  $\pi$  pode ser aproximado pelo somatório infinito:

$$\pi \approx 2 \left( 1 + \frac{1}{3} + \frac{1 \times 2}{3 \times 5} + \frac{1 \times 2 \times 3}{3 \times 5 \times 7} + \frac{1 \times 2 \times 3 \times 4}{3 \times 5 \times 7 \times 9} + \frac{1 \times 2 \times 3 \times 4 \times 5}{3 \times 5 \times 7 \times 9 \times 11} + \dots \right)$$

Escreva um programa para calcular um valor aproximado para  $\pi$  de acordo com a fórmula acima. O programa deve ler o número  $n$  de parcelas a serem usadas no cálculo do somatório. O programa deve testar se o valor de  $n$  é um inteiro positivo, solicitando que seja digitado novo valor de  $n$ , repetidamente, enquanto o valor lido não for válido.

Dois exemplos de execução do programa são apresentados a seguir.

**Exemplo 1:**

```
QUAL A QUANTIDADE DE PARCELAS (n)?: 0
ERRO: VALOR INVÁLIDO PARA n.
QUAL A QUANTIDADE DE PARCELAS (n)?: -1
ERRO: VALOR INVÁLIDO PARA n.
QUAL A QUANTIDADE DE PARCELAS (n)?: 2.2
ERRO: VALOR INVÁLIDO PARA n.
QUAL A QUANTIDADE DE PARCELAS (n)?: 1
```

VALOR APROXIMADO DE Pi = 2

**Exemplo 2:**

```
QUAL A QUANTIDADE DE PARCELAS (n)?: 3
VALOR APROXIMADO DE Pi = 2.93333
```

**Exemplo 3:**

```
QUAL A QUANTIDADE DE PARCELAS (n)?: 15
VALOR APROXIMADO DE Pi = 3.14157
```

#### Questão 14. (2016-01)

Escreva um programa para auxiliar uma pessoa a fazer compras no supermercado usando o seu *smartphone*. Inicialmente, a pessoa deve informar o valor que tem disponível para gastar na compra. A cada item a ser acrescentado ao carrinho, a pessoa digita o preço do produto. Se o preço for menor do que o saldo disponível, o item é colocado no carrinho e o saldo disponível é atualizado. Caso contrário, o item não é acrescentado ao carrinho e o saldo permanece o mesmo.

O programa deve interromper a compra quando o saldo disponível for igual a zero ou quando o consumidor digitar um preço de produto igual ou menor que zero. Ao final, o programa deve informar a valor da compra, o número de itens no carrinho e o saldo restante.

**Obs:** Você não precisa verificar se o valor do saldo inicial é válido, podendo supor que será digitado um valor positivo.

Dois exemplos de execução do programa são apresentados a seguir.

##### Exemplo 1:

```
Sistema de compras
-----
Digite o valor disponível para a compra: 25
Digite o valor do item: 12
Item 1 acrescentado ao carrinho
Saldo restante: 13.00
Digite o valor do item: 10
Item 2 acrescentado ao carrinho
Saldo restante: 3.00
Digite o valor do item: 3
Item 3 acrescentado ao carrinho
Saldo restante: 0.00
Digite o valor do item: 0
=====
Dirija-se ao caixa
Valor total da compra: R$ 25.00
Número de itens: 3
Saldo restante: R$ 0.00
```

##### Exemplo 2:

```
Sistema de compras
-----
Valor disponível para a compra: 30
Digite o valor do item: 10
Item 1 acrescentado ao carrinho
Saldo restante: 20.00
Digite o valor do item: 30
Saldo insuficiente, item não acrescentado ao carrinho
Saldo restante: 20.00
Digite o valor do item: 0
=====
Dirija-se ao caixa
Valor total da compra: R$ 10.00
Número de itens: 1
Saldo restante: R$ 20.00
```

**Questão 15.** (2016-01)

Escreva um programa que leia um valor inteiro positivo  $n$  e imprima um tabuleiro de jogo tal como ilustrado a seguir, para  $n = 12$ :

1	2	3	4	5	6	7	8	9	10	11	12
1	x										
2	#	+									
3	x	x	x								
4	#	+	#	+							
5	x	x	x	x	x						
6	#	+	#	+	#	+					
7	x	x	x	x	x	x	x				
8	#	+	#	+	#	+	#	+			
9	x	x	x	x	x	x	x	x			
10	#	+	#	+	#	+	#	+	#	+	
11	x	x	x	x	x	x	x	x	x	x	
12	#	+	#	+	#	+	#	+	#	+	+

Ou seja, as posições abaixo da diagonal principal, incluindo essa diagonal, são marcadas com **x**, nas linhas ímpares, ou com **#** e **+** alternados, nas linhas pares. Dois exemplos de execução do programa são apresentados a seguir.

**Exemplo 1:**

```
TABULEIRO
Dimensão do tabuleiro: 6
  1 2 3 4 5 6
1 x
2 #
3 x x x
4 # + # +
5 x x x x x
6 # + # + # +
```

**Exemplo 2:**

```
TABULEIRO
Dimensão do tabuleiro: 7
  1 2 3 4 5 6 7
1 x
2 #
3 x x x
4 # + # +
5 x x x x x
6 # + # + # +
7 x x x x x x x
```

**Questão 16.** (2016-02)

Uma imobiliária fez uma cotação de preços de imóveis em Ouro Preto e quer determinar qual é o imóvel com menor preço por metro quadrado. Você vai escrever um programa para realizar essa tarefa. O programa deve ler os seguintes dados:

1. O número de imóveis cujo preço foi cotado (você pode supor que sempre será digitado um valor inteiro maior que 0, não precisando testar se o valor digitado é válido) .
2. Para cada imóvel, o preço total (em reais) e a área do imóvel (em  $m^2$ )

O programa deve imprimir o número do imóvel de menor preço por  $m^2$ , o preço por  $m^2$  e o preço total desse imóvel. Um exemplo de execução do programa é mostrado a seguir.

**Exemplo 1:**

**COTAÇÃO DE IMÓVEIS**

**Quantidade de imóveis: 5**

**Preço do imóvel 1: 350000**

**Área do imóvel 1: 70**

**Preço do imóvel 2: 430000**

**Área do imóvel 2: 79**

**Preço do imóvel 3: 280000**

**Área do imóvel 3: 57**

**Preço do imóvel 4: 680000**

**Área do imóvel 4: 90**

**Preço do imóvel 5: 570000**

**Área do imóvel 5: 87**

**Imóvel de menor preço por m<sup>2</sup> = 3**

**Preço por m<sup>2</sup> = R\$ 4912.28**

**Preço total do imóvel = R\$ 280000.00**

**Questão 17. (2016-02)**

Uma determinada população de animais cresce a uma taxa anual constante  $r$ , isto é, para cada ano  $i > 0$ , a população  $N_i$  no final deste ano é dada por

$$N_i = N_{i-1} (1+r)$$

sendo  $N_{i-1}$  a população no final do ano anterior ( $i-1$ ).

Escreva um programa que leia o valor da taxa de crescimento ( $r$ ) e

1. Caso a taxa de crescimento  $r$  seja positiva, o programa deve ler a população inicial ( $N_0$ ) e imprimir uma tabela mostrando a população ao final de cada ano, até que a população atinja um valor maior ou igual a um milhão de indivíduos. Observe que a população em cada ano deve ser um número inteiro. Portanto, para calcular a população ao final de um dado ano, você deve multiplicar a população do ano anterior por  $(1+r)$  e tomar a parte inteira do resultado.
2. Caso a taxa de crescimento seja menor ou igual a 0, o programa deve exibir a mensagem “A população não irá crescer”

Dois exemplos de execução do programa são mostrados a seguir.

**Exemplo 1:**

**CRESCIMENTO POPULACIONAL**

**Taxa de crescimento: -0.4**

**A população não irá crescer**

**Exemplo 2:**

**CRESCIMENTO POPULACIONAL**

**Taxa de crescimento: 2.9**

**População inicial: 200**

**Ano População**

**0 200**

**1 780**

**2 3042**

**3 11863**

**4 46265**

**5 180433**

**6 703688**

**7 2744383**

**Questão 18.** (2016-02)

Escreva um programa que leia dois números inteiros positivos, **n1** e **n2**, e imprima uma figura tal como a mostrada a seguir, onde **n1** é o número de linhas e **n2** é o número de colunas da figura. No exemplo abaixo, **n1 = 5** e **n2 = 8**:

```
1 2 3 4 5 6 7 8
8 7 6 5 4 3 2 1
1 2 3 4 5 6 7 8
8 7 6 5 4 3 2 1
1 2 3 4 5 6 7 8
```

Um exemplo de execução do programa é mostrado a seguir.

**Exemplo 1:**

```
--FIGURA--
Digite o número de linhas: 6
Digite o número de colunas: 5

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

**Questão 19.** (2017-01)

Em uma empresa metalúrgica o lingote passa por um tratamento térmico visando a definição de sua composição final. Este tratamento é definido por uma curva de aquecimento conforme a expressão abaixo, em que o cálculo da temperatura atual (*i*) é feito com base na temperatura anterior (*i-1*) e é encerrado quando uma temperatura limite é atingida.

$$T_i = T_{i-1} + A * (T_{i-1} - T_p)$$

$T_i$  é o valor da temperatura no instante atual *i*,  $T_{i-1}$  é o valor da temperatura no instante anterior a *i*,  $A$  é a taxa de aquecimento e  $T_p$  é uma constante de perda de temperatura. Escreva um programa que leia a temperatura inicial  $T_0$ , a taxa de aquecimento  $A$ , a constante de perda  $T_p$ , a temperatura limite  $L$  e imprima uma tabela mostrando a temperatura em cada instante de tempo, tal como mostrado no exemplo abaixo, até que a temperatura limite seja ultrapassada. Adicionalmente, o programa deve imprimir a temperatura excedente e o instante *i*.

**Exemplo 1:**

```
Temperatura inicial: 200
Taxa de aquecimento: 0.5
Constante de perda: 50
Temperatura limite: 500
Tempo Temperatura
 0      200.00
 1      275.00
 2      387.50
 3      556.25
O excedente de temperatura é 56.25, obtida no instante 3.
```

**Questão 20.** (2017-01)

Um professor avalia seu alunos através de suas notas em várias atividades e a sua frequência. Para cada atividade o aluno recebe uma nota entre 0 e 10, ao final, o professor elimina a menor e a maior notas e calcula a média dentre as notas restantes. Implemente um programa que ajude o professor a avaliar seus alunos. Inicialmente o programa lê a quantidade de alunos e quantas atividades foram realizadas. Em seguida, para cada aluno, lê a frequência (de 0 a 100) cada uma das notas. Após definidas todas as notas, o programa imprime a nota obtida pelo aluno e o resultado (“**Aprovado!**”, “**Reprovado por Nota e Frequência!**”, “**Reprovado por Nota!**” ou “**Reprovado por Frequência!**”), considerando que a nota mínima para aprovação seja **6** e a frequência mínima seja **75**

**Exemplo 1:**

```
Quantidade de alunos: 1
Quantidade de avaliações: 4
```

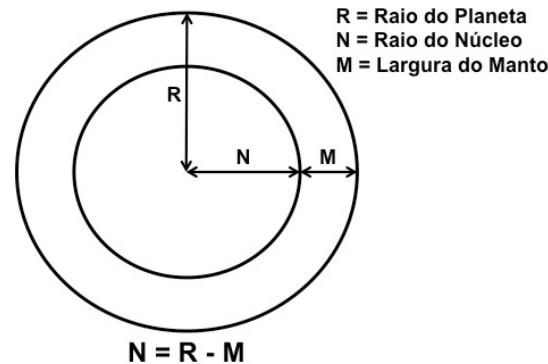
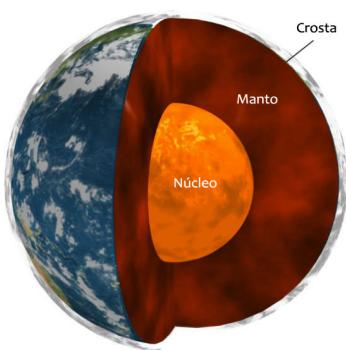
```
Aluno 1
Frequência do aluno: 80
Nota 1: 3
Nota 2: 5.5
Nota 3: 6.5
Nota 4: 4.5
Nota obtida: 5.0.
Reprovado por Nota!

Fim do programa.
```

**Questão 21.** (2017-02)

Um pesquisador está usando geotecnologia para estimar o raio do núcleo de um planeta. Para isso ele divide o planeta em três camadas: **crosta**, **manto** e **núcleo**. A crosta é ignorada e ele possui um aparelho que estima o raio do planeta e a largura do manto e, a partir destes números, ele calcula o raio estimado para o núcleo. A estimativa é obtida através de N leituras usando o aparelho e o resultado é o valor médio das estimativas para o raio do núcleo.

As figuras abaixo mostram uma representação de um planeta dividido em camadas e como calcular a estimativa do raio do núcleo a partir do raio do planeta e da largura do manto.



Escreva um programa Scilab que pergunte ao usuário o número de leituras e todos os valores lidos e calcule a estimativa do raio do planeta.

**Exemplo 1:**

```
Raio do Núcleo
Número de leituras: 3
Raio do planeta: 200
Largura do manto: 50
Raio do planeta: 202
Largura do manto: 51
Raio do planeta: 199
Largura do manto: 50
A estimativa do raio do núcleo é 150.00!
```

**Exemplo 2:**

```
Raio do Núcleo
Número de leituras: 2
Raio do planeta: 250
Largura do manto: 100
Raio do planeta: 245
Largura do manto: 105
A estimativa do raio do núcleo é 145.00!
```

**Questão 22. (2017-02)**

A Caixa Econômica Federal permite o comprometimento de no máximo **30%** do salário de seus clientes com empréstimos imobiliários. Desta forma, o banco precisa informar para os correntistas se o crédito para financiamento imobiliário está liberado ou não, baseado no valor das prestações do empréstimo (**P**) e no valor atual do salário (**S**) do cliente. Caso não esteja liberado, o banco precisa informar em quantos anos este cliente poderá pleitear um empréstimo neste valor, considerando que o salário sofre um reajuste (aumento) anual de apenas **6.7%**.

Escreva um programa Scilab que faça uma simulação para o atendimento de vários clientes solicitando, para cada cliente, o valor do salário atual (**S**) e, em seguida, o valor da prestação mensal fixa do empréstimo (**P**). O programa então informa se o crédito está liberado ou não. Caso não esteja, o programa calcula o tempo em anos necessário para que aquele cliente possa fazer o empréstimo, ou seja, quando a prestação for menor ou igual a **30%** do salário, considerando o reajuste anual de **6.7%**. O número de clientes simulados é informado pelo usuário no início da execução do programa.

**Exemplo 1:**

```
Simulação de empréstimos CEF
Informe a quantidade de clientes: 3
```

```
Cliente 1:
Salário atual: 5000
Prestação desejada: 1250
CRÉDITO LIBERADO!
```

```
Cliente 2:
Salário atual: 5000
Prestação desejada: 2300
CRÉDITO NEGADO!
```

Este empréstimo poderá ser feito somente daqui a 7 anos.

```
Cliente 3:
Salário atual: 1000
Prestação desejada: 350
CRÉDITO NEGADO!
```

Este empréstimo poderá ser feito somente daqui a 3 anos.

**Questão 23.** (2018-01)

Considere a série abaixo:

$$\frac{a^n}{1!} + \frac{(a \times 2)^{n-1}}{2!} + \frac{(a \times 3)^{n-2}}{3!} + \frac{(a \times 4)^{n-3}}{4!} + \dots + \frac{(a \times n)^1}{n!}$$

Por exemplo, para o caso de  $n = 5$ , temos então a série:

$$\frac{a^5}{1!} + \frac{(a \times 2)^4}{2!} + \frac{(a \times 3)^3}{3!} + \frac{(a \times 4)^2}{4!} + \frac{(a \times 5)^1}{5!}$$

Escreva um programa Scilab que calcule o somatório desta série com  $n$  parcelas seguindo o algoritmo:

1. Ler pelo teclado o valor de  $n$ . Este valor sempre será positivo e maior que zero. Não é necessário validar esse valor.
2. Ler pelo teclado o valor da constante  $a$ ;
3. Calcular o somatório e imprimir o resultado.
4. Não é permitida a utilização da função para o cálculo do fatorial (**factorial**).

As entradas e saídas de dados se comportam como os exemplos de execução abaixo.

**Exemplo 1:**

```
Qual o número de parcelas? 1
Qual o valor da constante a? 2
Somatório com 1 parcelas: 2
```

**Exemplo 2:**

```
Qual o número de parcelas ? 2
Qual o valor da constante a ? 3
Somatório com 2 parcelas: 12
```

**Exemplo 3:**

```
Qual o número de parcelas ? 4
Qual o valor da constante a ? 2
Somatório com 4 parcelas: 54.3333
```

**Questão 24.** (2018-01)

Um aluno da UFOP é responsável pelas compras mensais de supermercado de uma quantidade determinada de repúblicas de estudantes. Para isto, ele recebe de cada uma das repúblicas a lista de produtos que serão comprados.

Escreva um programa Scilab que simula as compras das repúblicas. O programa deve calcular o valor da compra de cada uma das repúblicas e o valor total de todas as compras efetuadas.

A seguir, um exemplo de execução do programa. Não é necessária a validação que qualquer entrada de dados.

**Exemplo 1:**

```
Informe a quantidade de repúblicas: 3

República número 1:
Quantos produtos na lista de compras: 2
Qual o valor do produto ? 15.00
Qual o valor do produto ? 12.00
Compras da Repúblca 1: R$ 27.00
```

```
República número 2:
Quantos produtos na lista de compras: 4
Qual o valor do produto ? 10.00
Qual o valor do produto ? 20.00
```

```
Qual o valor do produto ? 30.00
Qual o valor do produto ? 40.00
Compras da República 2: R$ 100.00
```

República número 3:

```
Quantos produtos na lista de compras: 5
Qual o valor do produto ? 1.00
Qual o valor do produto ? 2.00
Qual o valor do produto ? 3.00
Qual o valor do produto ? 4.00
Qual o valor do produto ? 5.00
Compras da República 3: R$ 15.00
```

**Total das compras: R\$ 142.00**

### Questão 25. (2018-01)

Escreva um programa para simular um terminal de banco. O programa deve iniciar lendo o saldo do usuário através de uma entrada do usuário, em seguida, o programa oferece uma série de operações a serem executadas, que são:

1. **Saldo**: apresenta o valor atual do saldo na tela.
2. **Retirada**: solicita o valor a ser debitado da conta, e se o saldo for suficiente, realiza a retirada e apresenta o saldo atualizado na tela. Caso contrário, imprime a mensagem: "Saldo insuficiente para esta retirada!".
3. **Depósito**: solicita o valor a ser creditado na conta, atualiza o valor do saldo e apresenta o saldo atualizado na tela.
4. **Encerrar**: o programa será encerrado.

O usuário deve fornecer um valor numérico, representando cada uma das operações, caso um valor inválido seja digitado, o programa deverá imprimir a mensagem de erro "Operação inválida!" na tela e solicitar uma nova escolha do usuário. Veja abaixo um exemplo de execução.

#### Exemplo 1:

```
Saldo inicial: 5350.00
=====
Escolha a operação (1-Saldo, 2-Retirada, 3-Depósito, 4-Encerrar): 1
Saldo em conta corrente: R$ 5350.00
=====
Escolha a operação (1-Saldo, 2-Retirada, 3-Depósito, 4-Encerrar): 3
Digite o valor do depósito: 8456.00
Saldo atualizado: R$ 13806.00
=====
Escolha a operação (1-Saldo, 2-Retirada, 3-Depósito, 4-Encerrar): 5
Operação inválida!
=====
Escolha a operação (1-Saldo, 2-Retirada, 3-Depósito, 4-Encerrar): 2
Digite o valor a ser debitado: 10000.00
Saldo atualizado: R$ 3806.00
=====
Escolha a operação (1-Saldo, 2-Retirada, 3-Depósito, 4-Encerrar): 2
Digite o valor a ser debitado: 5000.00
Saldo insuficiente para esta retirada!
=====
Escolha a operação (1-Saldo, 2-Retirada, 3-Depósito, 4-Encerrar): 4
```

### Questão 26. (2018-02)

Um estudante de Biologia quer “simular” uma série de crescimentos populacionais. Neste estudo, temos cada espécie uma população inicial  $N_0$ , uma taxa de crescimento  $r$ , que corresponde à taxa de natalidade da espécie e o seu nível de saturação  $K$ , que corresponde ao número máximo de indivíduos que o ambiente comporta. O estudo consiste em simular o crescimento de uma série de espécies distintas, sendo que para cada espécie são conhecidas as seguintes informações:

1. a população inicial  $N_0$ ,
2. a taxa de crescimento da população  $r$  e
3. o nível de saturação  $K$ .

Sabendo-se que o crescimento populacional é dado pela fórmula:  $N_i = N_{i-1} \times (1+r)$ , ou seja, o número de indivíduo da próxima geração ( $i$ ) é igual ao número de indivíduos da geração atual ( $i-1$ ) acrescido dos “novos” indivíduos, escreva um programa para:

1. ler total de espécies do estudo,
2. para cada espécie, fazer:
  - a. ler a população inicial  $N_0$ ,
  - b. ler a taxa de crescimento da população  $r$  e
  - c. ler o nível de saturação  $K$ .

Calcular e informar quantas gerações são necessárias para que a população de cada espécie ultrapasse o nível de saturação  $K$  e a população final.

#### Exemplo

```
Número de espécies: 3
Entre com os dados da Espécie 1
- População Inicial: 100
- Taxa de crescimento: 0.3
- Nível de saturação: 500
- São necessárias 7 gerações.
- População final: 627
Entre com os dados da Espécie 2
- População Inicial: 10
- Taxa de crescimento: 0.05
- Nível de saturação: 100
- São necessárias 48 gerações.
- População final: 104
Entre com os dados da Espécie 3
- População Inicial: 50
- Taxa de crescimento: 0.03
- Nível de saturação: 500
- São necessárias 78 gerações.
- População final: 501
```

### Questão 27. (2018-02)

O Imposto de Renda de Retido na Fonte (IRRF) é um valor descontado mensalmente do salário dos trabalhadores e repassado ao Governo Federal. Considere que o valor descontado do rendimento é calculado da seguinte maneira: Quem recebe até R\$ 1.900,00 não terá imposto retido na fonte, ou seja, não tem desconto algum no salário. Quem recebe de R\$ 1.900,01 até R\$ 2.800,00 mensais, paga uma alíquota de 7,5% de imposto e tem um desconto neste imposto de R\$ 140,00. Para quem recebe de R\$ 2.800,01 até R\$ 3.750,00, a alíquota é de 15% e desconto de R\$ 360,00. Para quem recebe de R\$ 3.750,01 até R\$ 4.700,00, a alíquota é de 22,5% e o desconto é de R\$ 650,00 e para quem recebe acima de R\$ 4.700,01, alíquota é de 27,5 e o desconto é de R\$ 870,00. **Obs.:** Nenhum trabalhador pode ganhar menos do que o salário mínimo, ou seja, menos do que R\$ 954,00. Caso isso ocorra, uma mensagem de erro, conforme exemplo de execução, deve ser impressa na tela.

Escreva um programa Scilab que faça o cálculo de uma série de salários digitados pelo usuário, informando o valor do imposto, o valor de desconto e o valor do salário líquido do trabalhador. O programa deve ser interrompido quando for digitado um valor menor ou igual a zero para o salário do trabalhador.

#### Exemplo

```
Digite o salário do trabalhador (<= 0 para fim): 1000
Imposto(-): R$ 0.00
Desconto(+): R$ 0.00
Sal. líquido: R$ 1000.00

Digite o salário do trabalhador (<= 0 para fim): 2000
Imposto(-): R$ 150.00
Desconto(+): R$ 140.00
Sal. líquido: R$ 1990.00

Digite o salário do trabalhador (<= 0 para fim): 10000
Imposto(-): R$ 2750.00
Desconto(+): R$ 870.00
Sal. líquido: R$ 8120.00

Digite o salário do trabalhador (<= 0 para fim): 500
O salário inválido, deve ser >= R$ 954.00

Digite o salário do trabalhador (<= 0 para fim): 4000
Imposto(-): R$ 900.00
Desconto(+): R$ 650.00
Sal. líquido: R$ 3750.00

Digite o salário do trabalhador (<= 0 para fim): 0
```

**Questão 28.** (2018-02)

O Campus Aberto da UFOP realizará uma gincana e precisa de um programa para computar a pontuação das duas equipes participantes e determinar a equipe vencedora da seguinte forma:

1. serão realizadas **N** provas, definidas por entrada do usuário;
2. cada equipe receberá uma nota entre 0 e 10 para cada prova (não é necessário validar);
3. a equipe com maior pontuação será a vencedora da prova, podendo haver empate;
4. para cada prova, deverá ser lida a nota de cada equipe e computada a equipe vencedora;
5. ao final informar qual foi a equipe que venceu o maior número de provas, ou se houve empate;
6. o programa imprime a pontuação de cada equipe e a equipe vencedora na tela.

**Exemplo 1**

```
Quantidade de provas: 5
==== Prova 1 ====
Pontuação da equipe 1: 5
Pontuação da equipe 2: 6
==== Prova 2 ====
Pontuação da equipe 1: 7.8
Pontuação da equipe 2: 7.6
==== Prova 3 ====
Pontuação da equipe 1: 5
Pontuação da equipe 2: 4
==== Prova 4 ====
Pontuação da equipe 1: 6
Pontuação da equipe 2: 7
==== Prova 5 ====
Pontuação da equipe 1: 3
Pontuação da equipe 2: 4
==== Resultados ===
A equipe 1 venceu 2 prova(s)
A equipe 2 venceu 3 prova(s)
Houve 0 empate(s) entre as equipes
Equipe 2 é a vencedora
```

**Exemplo 2**

```
Quantidade de provas: 3
==== Prova 1 ====
Pontuação da equipe 1: 5
Pontuação da equipe 2: 5
==== Prova 2 ====
Pontuação da equipe 1: 8
Pontuação da equipe 2: 6
==== Prova 3 ====
Pontuação da equipe 1: 4
Pontuação da equipe 2: 6
==== Resultados ===
A equipe 1 venceu 1 prova(s)
A equipe 2 venceu 1 prova(s)
Houve 1 empate(s) entre as equipes
Houve empate entre as equipes!
```