

# Lista de exercícios

Parte dos exercícios que operam sobre dados informados pelo usuário definem a forma como esses dados são recebidos: *na execução* indica que os dados serão solicitados ao usuário durante a execução do programa; *na chamada* indica que os dados serão passados como argumentos na chamada do programa pelo terminal. Outros exercícios que usam dados do usuário não especificam a forma de entrada dos dados; neste caso, qualquer das opções pode ser usada para este fim.

## 1 Variáveis e operações básicas

### Exercício 1.1 (`soma_expressao`)

Escreva um programa que exiba o resultado de  $2a \times 3b$ , onde  $a$  vale 3 e  $b$  vale 5.

### Exercício 1.2 (`soma_variaveis`)

Escreva um programa que define três variáveis inteiras e apresenta a soma delas.

### Exercício 1.3 (`area_triangulo`)

Escreva um programa que calcule a área  $A$  de um triângulo, cuja base  $b = 6$  e altura  $h = 5$ .

$$A = \frac{b \cdot h}{2}$$

### Exercício 1.4 (`aumento_salario`)

Escreva um programa que calcule um aumento de 23% em um salário de \$ 3600,00. Apresente o valor de aumento e o novo salário.

### Exercício 1.5 (`media_altura`)

Escreva um programa que receba (na execução) a altura de quatro pessoas e exiba em tela a média de altura do grupo.

### Exercício 1.6 (`metros_mili`)

Escreva um programa que leia um valor em metros (na execução) e o exiba convertido em milímetros.

### Exercício 1.7 (`converte_tempo`)

Escreva um programa que leia a quantidade de dias, horas, minutos e segundos do usuário (na chamada). Calcule o total em segundos.

### Exercício 1.8 (`desconto`)

Escreva um programa que leia o preço de uma mercadoria e o percentual de desconto (na chamada). Exiba o valor do desconto e o preço a pagar.

### Exercício 1.9 (`tempo_viagem`)

Escreva um programa que calcule o tempo de uma viagem de carro. Pergunte a distância a percorrer e a velocidade média esperada para a viagem.

### Exercício 1.10 (`converte_temperatura`)

Escreva um programa que converta uma temperatura dada em °C para °F, conforme fórmula abaixo.

$$F = \frac{9 \times C}{5} + 32$$

### Exercício 1.11 (`aluguel_carro`)

Escreva um programa que leia (na execução) a quantidade de km percorridos por um carro alugado pelo usuário, assim como a quantidade de dias pelos quais o carro foi alugado. Calcule o preço a pagar, sabendo que o carro custa \$ 60,00 por dia e \$ 0,15 por km rodado.

### Exercício 1.12 (`quilowatt`)

Sabe-se que o quilowatt de energia custa um quinto do salário mínimo. Escreva um programa que receba o valor do salário mínimo e a quantidade de quilowatts consumida por uma residência. Calcule e mostre:

- a) o valor de cada quilowatt;
- b) o valor da conta de energia elétrica dessa residência;
- c) o valor a ser pago com desconto de 15%.

### Exercício 1.13 (`opera_real`)

Escreva um programa que receba um número real (na chamada), encontre e mostre:

- a) a parte inteira desse número;
- b) a parte fracionária desse número;
- c) o arredondamento desse número para uma casa decimal;
- d) o arredondamento desse número para um número inteiro.

## 2 Estruturas condicionais

### Exercício 2.1 (`analisa_numeros`)

Escreva um programa que leia dois números (na execução) e informe qual o maior. No caso de serem iguais, o programa deve informar a igualdade.

### Exercício 2.2 (`multa`)

Escreva um programa que pergunte a velocidade do carro de um usuário. Caso ultrapasse 80 km/h, exiba uma mensagem dizendo que o usuário foi multado. Nesse caso, exiba o valor da multa, cobrando \$ 30,00 por km acima de 80 km/h.

### Exercício 2.3 (`maior_menor`)

Escreva um programa que leia três números (na execução) e imprima o maior e o menor.

### Exercício 2.4 (`aumento_salario_faixa`)

Escreva um programa que pergunte o salário do funcionário e calcule o valor do aumento. Para salários

superiores a \$ 1.250,00, calcule um aumento de 10%. Para os inferiores ou iguais, calcule um aumento de 15%.

#### Exercício 2.5 (preco\_viagem)

Escreva um programa que pergunte a distância que um passageiro deseja percorrer em km. Calcule o preço da passagem, cobrando \$ 0,50 por km para viagens de até 200 km; e \$ 0,45 por km para viagens mais longas.

#### Exercício 2.6 (operacao)

Escreva um programa que leia dois números (na execução) e pergunte qual operação o usuário deseja realizar. Deve ser possível calcular a soma (+), subtração (-), multiplicação (\*) e divisão (/). Exiba o resultado da operação solicitada.

- **Atenção:** o programa deve exibir a mensagem “valor inválido” caso o usuário tente uma divisão por zero.

#### Exercício 2.7 (emprestimo)

Escreva um programa para aprovar o empréstimo bancário para compra de uma casa. O programa deve perguntar o valor da casa a comprar, o salário e a quantidade de anos a pagar. O valor da prestação mensal não pode ser superior a 30% do salário. Calcule o valor da prestação como sendo o valor da casa a comprar dividido pelo número de meses a pagar.

#### Exercício 2.8 (energia\_eletrica)

Escreva um programa que calcule o preço a pagar pelo fornecimento de energia elétrica. Pergunte a quantidade de kWh consumida e o tipo de instalação: “R” para residências; “I” para indústrias; e “C” para comércios. Calcule o preço a pagar de acordo com a tabela a seguir (o preço é dado por kWh).

Tipo	Faixa (kWh)	Preço
Residencial	Até 500	\$ 0,40
	Acima de 500	\$ 0,65
Comercial	Até 1000	\$ 0,55
	Acima de 1000	\$ 0,60
Industrial	Até 5000	\$ 0,57
	Acima de 5000	\$ 0,68

#### Exercício 2.9 (triangulo\_lados)

Escreva um programa que receba (na chamada) o comprimento de três retas e determine se essas retas podem formar um triângulo. Lembre-se de que qualquer lado do triângulo tem que ser menor que a soma dos dois outros lados.

#### Exercício 2.10 (triangulo\_angulos)

Escreva um programa que receba (na chamada) os valores dos ângulos internos de um triângulo. O programa deve informar caso os valores informados não podem ser associados aos ângulos internos de um triângulo. Caso contrário, o programa deve informar se o triângulo é isósceles, se ele é um triângulo retângulo, ou nenhum deles.

#### Exercício 2.11 (nota\_ponderada)

A nota final de um estudante é calculada a partir de três avaliações: um trabalho de laboratório, uma

prova semestral e um exame final. Os pesos dessas notas são 2, 3 e 5, respectivamente. Escreva um programa que receba as três notas (na execução), calcule e mostre a média ponderada e o conceito final do aluno. conforme a tabela abaixo.

Média ponderada	Conceito
[8,0, 10,0]	A
[7,0, 8,0]	B
[6,0, 7,0]	C
[5,0, 6,0]	D
[0,0, 5,0]	E

**Exercício 2.12 (ordem\_crescente)**

Escreva um programa que receba três números (na chamada) e mostre-os em ordem crescente. Suponha que os três números sejam diferentes.

**Exercício 2.13 (insere\_numero)**

Escreva um programa que receba três números obrigatoriamente em ordem crescente e um quarto número que não siga essa regra (na chamada). Mostre, em seguida, os quatro números em ordem decrescente. Suponha que os quatro números sejam diferentes.

**Exercício 2.14 (par\_impar)**

Escreva um programa que receba um número (na execução) e verifique se é par ou ímpar.

**Exercício 2.15 (apresenta\_numeros)**

Escreva um programa que receba quatro valores: I, A, B, C. Desses valores, I é inteiro e positivo, A, B e C são reais. Escreva os números A, B e C obedecendo à tabela a seguir. Suponha que o valor digitado para I seja sempre um valor válido, ou seja, 1, 2 ou 3, e que os números digitados sejam diferentes um do outro.

Valor de I	Forma a escrever
1	A, B e C em ordem crescente
2	A, B e C em ordem decrescente
3	O maior fica entre os dois números

**Exercício 2.16 (recursos\_humanos)**

Escreva um programa que apresente o menu a seguir, permite ao usuário escolher a opção desejada, receba os dados necessários para executar a operação e mostre o resultado. Verifique a possibilidade de opção inválida e não se preocupe com restrições, como salário negativo.

Menu de opções:

1. Imposto
2. Novo salário
3. Classificação

**Na opção 1:** receber o salário do funcionário, calcular e mostrar o valor do imposto usando as regras a seguir.

Salário	Imposto
Menor que \$ 500,00	5%
De \$ 500,00 (inclusive) a \$ 850,00 (inclusive)	10%
Acima de \$ 850,00	15%

**Na opção 2:** receber o salário do funcionário, calcular e mostrar o valor do novo salário, usando as regras a seguir.

Salário	Aumento
Maior que \$ 1.500,00	\$ 25,00
De \$ 750,00 (inclusive) a \$ 1.500,00 (inclusive)	\$ 50,00
De \$ 450,00 (inclusive) a \$ 750,00	\$ 75,00
Menor que \$ 450,00	\$ 100,00

**Na opção 3:** receber o salário do funcionário e mostrar sua classificação usando a tabela a seguir.

Salário	Classificação
Até \$ 700,00 (inclusive)	Mal remunerado
Maior que \$ 700,00	Bem remunerado

### 3 Estruturas de repetição

#### Exercício 3.1 (multiplos)

Escreva um programa para escrever os primeiros 100 múltiplos de um número informado pelo usuário (na chamada).

#### Exercício 3.2 (multiplicacao)

Escreva um programa que leia dois números (na chamada) e imprima seu produto, sem uso do operador de multiplicação (\*).

#### Exercício 3.3 (divisao\_resto)

Escreva um programa que leia dois números (na chamada) e imprima a divisão inteira do primeiro número pelo segundo, e o resto da divisão. Não use os operadores de divisão e resto (/ , // e %).

#### Exercício 3.4 (juros)

Escreva um programa que receba (na chamada) o valor de um depósito inicial e a taxa de juros de uma aplicação. O programa deve informar o montante em cada mês dos primeiros dois anos de rendimento.

#### Exercício 3.5 (juros\_deposito)

Escreva um programa que receba (na chamada) o valor de um depósito inicial e a taxa de juros de uma

aplicação. A cada mês, o programa deve ler o valor do depósito daquele mês (na execução), e informar o montante em cada mês do primeiro ano de rendimento.

**Exercício 3.6 (leitura\_numeros)**

Escreva um programa que leia valores inteiros digitados pelo usuário (na execução). O programa deve ler números até que o usuário digite 0 (zero), e ao final apresentar a quantidade de números digitados, bem como a soma e a média aritmética.

**Exercício 3.7 (registradora)**

Escreva um programa para controlar uma pequena máquina registradora. Você deve solicitar ao usuário que digite o código do produto e a quantidade comprada. Use a tabela de códigos abaixo para obter o preço de cada produto.

Código	Preço
1	0,50
2	1,00
3	4,00
5	7,00
9	8,00

Seu programa deve exibir o total das compras depois que o usuário digitar 0. Qualquer outro código deve gerar a mensagem de erro “Código inválido”.

**Exercício 3.8 (n\_primos)**

Escreva um programa que leia (na chamada) um número  $n$  e imprima os  $n$  primeiros números primos.

**Exercício 3.9 (calcula\_e)**

Escreva um programa que leia (na chamada) um número  $n$  e calcule o valor de  $E$ , conforme a equação abaixo.

$$E = \sum_{i=0}^n \frac{1}{i!} = 1 + \frac{1}{1!} + \frac{1}{2!} + \dots + \frac{1}{n!}$$

**Exercício 3.10 (estatisticas\_transito)**

Foram coletados dados sobre acidentes de trânsito em cinco capitais brasileiras. Foram obtidas as seguintes informações: (a) código da cidade; (b) número de veículos de passeio; (c) número de acidentes de trânsito com vítimas.

Escreva um programa que leia os dados coletados (na execução) e informe:

1. o maior e menor índices de acidentes de trânsito e a quais cidades pertencem;
2. a média de veículos nas cinco cidades juntas;
3. a média de acidentes de trânsito nas cidades com menos de 2.000 veículos de passeio.

**Exercício 3.11 (fibonacci)**

escreva um programa que leia um inteiro  $n$  (na chamada) e escreva os  $n$  primeiros termos da sequência de Fibonacci.

### Exercício 3.12 (futebol)

Em um campeonato de futebol existem cinco times e cada um possui onze jogadores. Escreva um programa que receba a idade, o peso e a altura de cada jogador (na execução), calcule e mostre:

- a quantidade de jogadores com idade inferior a 18 anos;
- a média das idades dos jogadores de cada time;
- a média das alturas de todos os jogadores do campeonato;
- o percentual de jogadores com mais de 80 kg entre todos os jogadores do campeonato.

### Exercício 3.13 (soma\_pares)

Escreva um programa que leia um número não determinado de pares de valores  $(m, n)$ , todos inteiros positivos, um par de cada vez, e que calcule e mostre a soma de todos os números inteiros entre  $m$  e  $n$  (inclusive). A digitação de pares terminará quando  $m$  for maior ou igual a  $n$ .

### Exercício 3.14 (salario\_professores)

Escreva um programa que leia a classe (A ou B) e o número de horas/aula dadas mensalmente pelos professores de uma universidade, sabendo-se que cada hora/aula vale \$146,00. Emita uma listagem contendo o salário bruto e o salário líquido (levando em consideração os descontos detalhados abaixo) de todos os professores. Mostre também a média dos salários líquidos de professores das classes A e B (separadamente). Considere que:

- o desconto para professores das classes A e B é 10% e 5%, respectivamente;
- as informações terminarão quando for lida uma classe diferente de A e B.

## 4 Listas (vetores)

### Exercício 4.1 (soma\_posicoes)

Escreva um programa que leia uma lista com 10 elementos e dois valores  $x$  e  $y$ , correspondentes a duas posições da lista. O programa deve escrever a soma dos valores encontrados nas posições  $x$  e  $y$  da lista.

### Exercício 4.2 (lista\_impares\_primos)

Escreva um programa que crie uma lista contendo os 20 primeiros números ímpares. Ao final, exiba a lista em tela. Repita o processo para os 20 primeiros números primos.

### Exercício 4.3 (troca\_elementos)

Escreva um programa que leia uma lista de 16 posições. Troque os 8 primeiros elementos pelos 8 últimos e vice-versa. Faça essa operação sem usar listas auxiliares.

### Exercício 4.4 (busca)

Escreva um programa que leia uma lista com valores digitados pelo usuário, em seguida, um valor  $x$  qualquer. O programa deve fazer uma busca pelo valor  $x$  na lista, informando a (primeira) posição em que foi encontrado ou se não foi encontrado.

### Exercício 4.5 (menor\_maior)

Escreva um programa que, dada uma lista com 15 números inteiros, identifique e exiba em tela seu menor valor e seu maior valor.

**Exercício 4.6 (negativo\_zero)**

Escreva um programa que, dada uma lista com 12 números inteiros, atribua 0 para todas as posições que contenham valores negativos.

**Exercício 4.7 (transferencia)**

Escreva um programa que leia duas listas  $A$  e  $B$  com o mesmo tamanho, e transfira seus elementos para uma terceira lista  $C$ . Essa lista conterá, nas posições pares os valores das posições correspondentes da lista  $A$ , e nas posições ímpares os valores das posições correspondentes da lista  $B$ .

**Exercício 4.8 (acumula)**

Escreva um programa que leia uma lista com  $n$  posições e acumule em cada posição  $i \in [0, n - 1]$  a soma dos valores armazenados em todas as posições  $j \leq i$ . Ao final, exiba a lista obtida em tela.

**Exercício 4.9 (letras)**

Escreva um programa que leia duas listas com 5 letras cada. Verifique e escreva se ambas possuem o mesmo conteúdo. Considere que uma lista tem o mesmo conteúdo que a outra se contiverem as mesmas letras, não importando a ordem em que aparecem em ambas as listas.

**Exercício 4.10 (repeticao)**

Escreva um programa que leia uma lista com números inteiros e verifique se existem valores repetidos. Para cada valor repetido, escreva o valor e quantas vezes ele se repete no vetor.

**Exercício 4.11 (compactar)**

Escreva um programa que leia uma lista com 15 inteiros não negativos e a compacte. A compactação consiste em deixar todos os valores iguais a 0 nas últimas posições da lista.

**Exercício 4.12 (veiculos)**

Escreva um programa que preencha uma lista com os modelos de cinco veículos (e.g., Fusca, Uno, Brasília). Preencha outra lista com o consumo desses veículos, ou seja, quantos quilômetros cada um deles faz com um litro de combustível. Calcule e mostre:

- o modelo do veículo mais econômico;
- quantos litros de combustível cada veículo consome para percorrer uma distância de 1.000 km.

**Exercício 4.13 (aumento)**

Escreva um programa que preencha três listas com cinco posições cada. A primeira lista receberá os nomes de cinco funcionários. A segunda e terceira listas receberão, respectivamente, o salário e o tempo de serviço de cada um. Mostre um primeiro relatório apenas com os nomes dos funcionários que não terão aumento. Mostre um segundo relatório apenas com os nomes e os novos salários dos funcionários que terão aumento.

Sabe-se que os funcionários que terão direito ao aumento são aqueles que possuem tempo de serviço superior a cinco anos ou salário inferior a \$ 2000,00. Sabe-se ainda que, se o funcionário satisfizer às duas condições anteriores, tempo de serviço e salário, o aumento será de 35%. Para o funcionário que satisfizer apenas à condição de tempo de serviço, o aumento será de 27%. Para aquele que satisfizer apenas à condição de salário, o aumento será de 15%.

**Exercício 4.14 (lista\_primos)**

Escreva um programa que leia uma lista com 15 números naturais. Crie uma segunda lista que contenha todos os números primos da lista anterior. Exiba ambas as listas em tela.



## 5 Funções

Todos os exercícios desta seção devem ser resolvidos usando funções, mesmo quando não especificado no enunciado. Procure testar as funções implementadas com diferentes valores para seus parâmetros, a fim de verificar sua correção.

### Exercício 5.1 (`volume_esfera`)

Escreva uma função que receba por parâmetro o raio  $r$  de uma esfera e retorne seu volume, dado por  $v = 4\pi r^3/3$ .

### Exercício 5.2 (`media`)

Escreva uma função que receba as 3 notas de um aluno por parâmetro e uma letra, retornando a média resultante. Se a letra for “A”, a função calcula a média aritmética das notas do aluno, se for “P”, calcula sua média ponderada (pesos 5, 3 e 2).

### Exercício 5.3 (`bhaskara`)

Escreva uma função que receba os coeficientes de uma função quadrática e calcule suas raízes usando a fórmula de Bhaskara.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

### Exercício 5.4 (`triangulos`)

Escreva uma função que receba três valores reais  $x$ ,  $y$  e  $z$  e verifica se esses valores podem ser os comprimentos dos lados de um triângulo. Em caso positivo, retorna o tipo de triângulo formado. Para que  $x$ ,  $y$  e  $z$  formem um triângulo é necessário que a seguinte propriedade seja satisfeita: o comprimento de cada lado de um triângulo é menor do que a soma do comprimento dos outros dois lados. Para identificar o tipo de triângulo formado, a função deve seguir as seguintes definições:

- triângulo equilátero: os comprimentos dos 3 lados são iguais;
- triângulo isósceles: os comprimentos de 2 lados são iguais;
- triângulo escaleno: os comprimentos dos 3 lados são diferentes.

Leia os comprimentos dos lados de um número indefinido de triângulos na chamada do programa. Usando a função implementada, mostre em tela o tipo de cada triângulo, ou se não trata-se de um triângulo, bem como a quantidade de triângulos de cada tipo. Lembre-se que `sys.argv` é uma lista e seu tamanho (i.e. `len(sys.argv)`) é dado pela quantidade de parâmetros informados na chamada do programa.

### Exercício 5.5 (`triangulos_estendido`)

Modifique o programa do exercício “`triangulos`” para mostrar o perímetro de cada triângulo informado, bem como o perímetro total e médio dos triângulos. Implemente as operações necessárias em funções para facilitar a implementação e melhor estruturar seu programa.

### Exercício 5.6 (`volumes`)

Escreva um programa que, dado um tipo de forma geométrica, calcule e escreva seu volume. Leia os dados necessários na chamada e implemente funções para calcular o volume das seguintes formas ( $s$  é o comprimento de um lado,  $l$  é a largura,  $c$  é o comprimento,  $h$  é a altura e  $r$  é o raio):

- Cubo:  $V = s^3$

- Paralelepípedo:  $V = l \cdot c \cdot h$
- Cilindro:  $V = \pi \cdot r^2 \cdot h$
- Esfera:  $V = 4\pi r^3/3$
- Cone:  $V = \pi \cdot r^2 \cdot h/3$

### Exercício 5.7 (perfeito)

Escreva um programa que verifique se um número é perfeito. Um valor é dito perfeito quando ele é igual à soma dos seus divisores excetuando ele próprio (e.g., 6 é perfeito, pois  $6 = 1 + 2 + 3$ , que são seus divisores). Além de informar se o número é perfeito, o programa deve exibir sua lista de divisores.

### Exercício 5.8 (massa)

Um determinado material radioativo perde metade de sua massa a cada 50 segundos. Dada a massa inicial, em gramas, fazer um programa que determine o tempo necessário para que essa massa se torne menor do que 0,5 g. Escreva a massa inicial, a massa final e o tempo calculado em horas, minutos e segundos.

### Exercício 5.9 (brinquedos)

Um distribuidor de brinquedos fez um acordo de compra de um lote de brinquedos pequenos embalados em caixas de formato retangular de tamanhos variados. Ele pretende reembalar esses brinquedos em esferas de plástico e revendê-las como pacotes surpresa. As esferas são fornecidas com quatro diâmetros diferentes: 10, 15, 20 e 25 cm. Para pedir as esferas ele precisa informar quantas unidades de cada diâmetro necessita. Sabendo que a diagonal maior ( $D$ ) de uma caixa retangular com dimensões  $A$ ,  $B$ , e  $C$  é dada por  $D^2 = A^2 + B^2 + C^2$ , escreva um programa que determine o número de esferas, de cada tamanho e total, necessárias para embalar o lote de brinquedos. Determine também a quantidade de caixas que possuem diagonal maior que o diâmetro da maior esfera. As dimensões das caixas retangulares são lidas através do teclado. Um valor negativo ou nulo para a primeira dimensão lida servirá como marca final de introdução de dados.

## 6 Manipulação de strings

### Exercício 6.1 (conta\_vogais)

Escreva um programa que receba (na chamada) uma frase e conte quantas vogais existem nela. Escreva duas funções para implementar essa operação. A primeira função deve contar as vogais percorrendo e avaliando cada caractere da string (como se fosse uma lista). A segunda deve fazer uso da função `count`. Ao final, o programa deve exibir a frase em tela e informar o número de ocorrências de cada vogal, bem como o número total de vogais.

Aproveite uma das funções anteriores para informar ainda quantos caracteres diferentes de vogal há na frase (somente a quantidade).

### Exercício 6.2 (conta\_palavras)

Escreva um programa que receba (na execução) uma frase e conte quantas palavras existem nela. Para isso, primeiro remova os espaços em branco no início e no fim da frase. Substitua ocorrências de vários espaços por um espaço único, e então use a quantidade de espaços em branco na frase para determinar o número de palavras.

### Exercício 6.3 (letras\_repetidas)

Escreva um programa que receba (na execução) uma frase e mostre as letras repetidas na frase. O

programa deve mostrar a letra (que aparece pelo menos duas vezes na frase) e o número de ocorrências correspondente.

#### Exercício 6.4 (criptografia)

Escreva uma função para criptografar uma mensagem dada pelo usuário. A criptografia troca as vogais por outros caracteres conforme as seguintes regras: A → ∗; E → \$; I → #; O → &; U → %.

Escreva uma nova função para decriptografar uma mensagem dada pelo usuário, seguindo as mesmas regras acima.

#### Exercício 6.5 (combina)

Escreva um programa que receba duas frases e gere uma terceira que represente a combinação das palavras das duas frases recebidas. Por exemplo:

Frase 1: *Hoje está um belo dia*

Frase 2: *Talvez chova amanhã*

Saída: *Hoje talvez está chova um amanhã belo dia*

#### Exercício 6.6 (ordena)

Escreva um programa que receba uma frase e a exiba com as palavras em ordem alfabética. Por exemplo:

Frase: *A informática está em constante evolução*

Saída: *A constante em está evolução informática*

#### Exercício 6.7 (filtro)

Escreva um programa que receba uma mensagem e a classifique segundo as regras a seguir (note que uma mensagem pode ter mais de uma classificação ao mesmo tempo):

- *Estudos*, caso contenha pelo menos uma das palavras *universidade*, *curso*, *disciplina* ou *professor*.
- *Trabalho*, caso contenha pelo menos duas das palavras *empresa*, *cargo*, *reunião*, *documento*, *férias* ou *salário*.
- *Outros*, caso não seja classificada como nenhum dos tipos anteriores.

#### Exercício 6.8 (mouse)

Escreva um programa que receba uma frase e, a cada ocorrência da palavra *teclado*, insira o texto *ou mouse* logo após. Por exemplo:

Frase: *A informática está em constante evolução*

Saída: *A constante em está evolução informática*

#### Exercício 6.9 (login)

Escreva um programa que receba um nome e gere como saída o nome digitado e seu login. Lembre-se de respeitar as letras minúsculas e maiúsculas, já que o login será sempre com letras minúsculas. A regra para geração do login é: a primeira letra do nome e, caso exista apenas um sobrenome, deve-se acrescentá-lo; caso existam dois sobrenomes, deve-se gerar a primeira letra do nome, mais a primeira letra do primeiro sobrenome, seguido do último sobrenome; caso existam três ou mais sobrenomes, deve-se proceder como na situação anterior, considerando o nome, o primeiro sobrenome e o último sobrenome. Exemplos:

Nome: Pedro Hansdorf

Login: phansdorf

Nome: Robson Soares Silva

Login: rssilva

Nome: Jaqueline Oliveira Fernandes Rodrigues

Login: jorodrigues

### Exercício 6.10 (senha)

Escreva um programa que leia (na chamada) a senha desejada pelo usuário e seu primeiro nome, e então valide a senha conforme as seguintes regras: a senha não pode iniciar ou terminar com o nome do usuário; caso o nome apareça no meio da senha, ela deve incluir pelo menos 4 outros caracteres, incluindo ao menos um caractere numérico.

### Exercício 6.11 (funcionarios)

Escreva um programa que leia uma string (na chamada) contendo os dados dos funcionários de uma empresa. Os dados são o nome, a idade e o salário de cada funcionário e estão organizados da seguinte forma: nome, idade e salário são separados com vírgula, e dados de diferentes funcionários são separados com ponto e vírgula. Um exemplo de dados de dois funcionários é “*Maria,34,7600;João,48,5200*”. Após ler os dados de todos os funcionários, seu programa deve exibir um resumo contendo a menor e maior idades, bem como a idade média dos funcionários, o menor e maior salários, bem como o salário médio dos funcionários. O programa também deve apresentar uma tabela com os dados de todos os funcionários. O formato de saída esperado está especificado abaixo.

```
Menor idade.....34
Maior idade.....48
Idade média.....41
Menor salário.....5200.00
Maior salário.....7600.00
Salário médio.....6400.00
```

```
-----
Nome    Idade    Salário
-----
Maria    34        7600.00
João     48        5200.00
-----
```

## 7 Matrizes

### Exercício 7.1 (menor\_maior)

Escreva um programa que defina uma matriz de valores inteiros, cujas dimensões são 5 linhas e 5 colunas ( $5 \times 5$ ). O programa deve encontrar e exibir o menor e maior valores armazenados na matriz.

### Exercício 7.2 (soma\_matrizes)

Escreva um programa que defina duas matrizes  $A$  e  $B$  de valores inteiros com dimensões  $5 \times 5$ . O programa

deve fazer o somatório dos elementos de correspondentes de cada matriz, ou seja  $a_{i,j} + b_{i,j}$ , e apresentar os resultados em tela.

**Exercício 7.3 (soma\_diag)**

Escreva um programa que defina uma matriz de valores inteiros com dimensões  $8 \times 8$ . O programa deve fazer o somatório dos elementos da diagonal principal e armazenar em uma variável  $d$ , e fazer o somatório dos elementos da diagonal secundária, armazenando em uma variável  $d'$ . Ao final, mostrar o resultado de  $d - d'$ .

**Exercício 7.4 (soma\_diag)**

Escreva um programa que leia uma matriz quadrada de números inteiros com dimensões  $n \times n$  e conte quantos números primos estão contidos na matriz.

**Exercício 7.5 (transposta)**

Escreva um programa que leia uma matriz quadrada de números inteiros com dimensões  $n \times n$  e apresente a matriz transposta correspondente. A matriz transposta consiste na troca das linhas pelas colunas.

Exemplo:

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix} \quad A^t = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{bmatrix}$$

**Exercício 7.6 (simetrica)**

Escreva um programa que leia uma matriz de números inteiros e informe se a matriz é simétrica. Uma matriz é simétrica quando ela é quadrada e seus elementos acima da diagonal principal são iguais aos seus elementos abaixo da diagonal principal, i.e.  $a_{i,j} = a_{j,i}$ . A matriz transposta de uma matriz simétrica é igual a ela mesma, i.e.  $A = A^t$ . Um exemplo:

$$A = \begin{bmatrix} 1 & 3 & -1 & 4 \\ 3 & 7 & 5 & 1 \\ -1 & 5 & 3 & 3 \\ 4 & 1 & 3 & 1 \end{bmatrix}$$

**Exercício 7.7 (zeros)**

Escreva um programa que defina uma matriz de números inteiros com dimensões  $n \times m$  quaisquer e conte quantas linhas e quantas colunas são zeradas (i.e. todos seus elementos são iguais a zero).

**Exercício 7.8 (triangular\_superior)**

Escreva um programa que defina uma matriz quadrada de números inteiros e determine se ela é uma matriz triangular superior, ou seja, se todos os valores abaixo da diagonal principal são nulos (iguais a zero).

**Exercício 7.9 (notas)**

Escreva um programa para ajudar no cálculo das notas dos estudantes de um curso. Para isso, use uma matriz onde a primeira linha armazena o nome de cada estudante, e as demais linhas suas notas. Todos os

estudantes terão o mesmo número de notas. Por exemplo, a matriz abaixo mostra três notas dos alunos *Maria* (8.0, 7.5, 9.8), *Ana* (8.5, 9.0, 10.0) e *João* (5.1, 8.2, 6.0).

Maria	Ana	João
8.0	8.5	5.1
7.5	9.0	8.2
9.8	10.0	6.0

O programa deve ler o número de notas e os dados dos estudantes a partir do usuário até que ele digite um valor nulo para o nome (string vazia). Ao final, o programa deve apresentar a média (aritmética) de cada estudante, e mostrar os estudantes com melhor e pior desempenhos.

### Exercício 7.10 (placar)

Considere a seguinte tabela com os resultados de  $n$  jogos entre Palmeiras e Corinthians.

Palmeiras	Corinthians
4	0
2	1
0	0
8	1
2	3
3	0
2	2

Escreva um programa que leia os nomes das equipes e os resultados de  $n$  jogos, armazenando em uma matriz de  $n$  linhas e duas colunas. O programa deve calcular a pontuação total de cada equipe, sendo 3 pontos para cada vitória e 1 ponto para cada empate. Ao final, informe a equipe com melhor desempenho, ou se elas estão empatadas.

### Exercício 7.11 (estoque)

Escreva um programa para auxiliar na gestão de estoque de uma rede de lojas de informática. A rede controla os produtos vendidos e uma quantidade mínima de estoque para todas as lojas. Use listas (vetores) para armazenar a lista de produtos e os estoques mínimos, conforme os exemplos abaixo.

`produtos = ['monitor', 'tablet', 'impressora', 'teclado', 'mouse']`

`minimo = [20, 25, 15, 80, 120]`

Use uma matriz de  $n$  linhas por  $m$  colunas para armazenar os estoques de cada um dos  $m$  produtos para cada uma das  $n$  lojas, conforme o exemplo abaixo.

`estoques =`

30	38	20	96	145
18	59	18	81	138
55	21	40	71	220
36	25	21	98	150
12	30	21	112	127

O programa deve ler as listas *produtos* e *minimo*, bem como a matriz *estoques*, e verificar o estoque de cada produto em cada uma das lojas, identificando a ocorrência de uma das seguintes possíveis situações em cada loja:

1. todos os produtos estão de acordo com o estoque mínimo. Neste caso, é impressa uma mensagem informando tal situação; ou

2. são impressos os produtos cujo estoque está abaixo do estoque mínimo e a quantidade que falta para atingir esse estoque.