

## Comandos de decisão e quebras em comandos de repetição

1. Faça um programa que receba dois números inteiros e imprima o maior deles, se por acaso os dois números forem iguais, imprima a mensagem “Numeros iguais”.
2. Faça um programa que receba um número inteiro e imprima se o número é par ou ímpar.
3. Ler 4 números inteiros e calcular a soma dos que forem par.
4. Ler três valores e determinar o maior entre eles.
5. Leia um número fornecido pelo usuário. Se esse número for positivo, calcule a raiz quadrada do número. Se o número for negativo, mostre uma mensagem dizendo que o número é inválido.
6. Faça um programa que leia 3 notas, verifique se as notas são válidas e exiba na tela a média destas notas com duas casas decimais. Uma nota válida deve ser, obrigatoriamente, um valor entre 0.00 e 10.00, onde caso a nota não possua valor válido, este fato deve ser informado ao usuário e o programa termina.
7. Leia o salário de um trabalhador e o valor da prestação de um empréstimo. Se a prestação for maior que 20% do salário então imprima: "Emprestimo nao concedido", caso contrário imprima: "Emprestimo concedido".
8. Faça um programa que receba dois números e execute as operações listadas a seguir de acordo com a escolha do usuário:
  - 1: Média entre os números digitados
  - 2: Diferença do maior pelo menor
  - 3: Produto entre os números digitados
  - 4: Divisão do primeiro pelo segundoSe a opção digitada for inválida, mostrar uma mensagem de erro e terminar a execução do programa. Dica do Brother: Na operação 4 o segundo número deve ser diferente de 0.
9. Ler três inteiros e imprimi-los na tela em ordem crescente.
10. Escreva um programa que leia um inteiro entre 1 e 7 e imprima o dia da semana correspondente a este numero. Isto é, domingo se 1, segunda-feira se 2, e assim por diante.
11. Faça um programa que simule uma calculadora com as 4 operações básicas. O usuário digita o primeiro número, escolhe a operação e em seguida digita o segundo número, exatamente nessa ordem. O programação deve mostrar o resultado da operação.
12. Faça um programa que leia um valor digitado pelo usuário e imprima os resultados com base na tabela.
  - 1: Bom Dia - Boa Tarde - Boa Noite
  - 2: Por Favor :) - Com Licenca :D - Muito Obrigado ;)
  - 3: Por Gentileza - Voce poderia - Desculpe
  - 4: Boa Sorte - Tenha Fe
  - Outro: Estudar vale muito a pena!
13. Calcule as raízes da equação do 2º grau. Lembrando que:  $x = \frac{-b \pm \sqrt{\Delta}}{(2a)}$ , onde  $\Delta = b^2 - 4ac$ , e  $ax^2 + bx + c = 0$  representa uma equação do 2º grau. A variável a tem que ser diferente de zero. Caso seja igual, imprima a mensagem "Nao eh equacao do 2o grau".
  - Se  $\Delta < 0$  , não existe raiz real. Imprima a mensagem “Nao existe raiz real”.
  - Se  $\Delta = 0$  , existe uma raiz real. Imprima a raiz e a mensagem “Raiz unica”.
  - Se  $\Delta > 0$  , Imprima as duas raízes reais.
14. Faça um programa que apresente um menu de opções para o cálculo das seguintes operações entre dois números:
  - adição (opção 1)
  - subtração (opção 2)
  - multiplicação (opção 3)
  - divisão (opção 4)
  - saída (opção 5)

O programa deve possibilitar ao usuário a escolha da operação desejada, a exibição do resultado e a volta ao menu de opções. O programa só termina quando for escolhida a opção de saída (opção 5).

15. Escreva um programa que leia um número inteiro e calcule a soma de todos os divisores desse número, com exceção dele próprio. Exemplo: a soma dos divisores de 66 é  $1 + 2 + 3 + 6 + 11 + 22 + 33 = 78$ .
16. Elabore um programa que faça leitura de vários números inteiros, até que se digite um número negativo. O programa tem que retornar o maior e o menor dos números não-negativos lidos.
17. Faça um programa que leia um conjunto não determinado de valores, um de cada vez, e escreva, para cada um dos valores lidos, o quadrado, o cubo e a raiz quadrada. Finalize a entrada de dados com um valor negativo ou zero.
18. Faça um função que receba a data atual (dia, mês e ano em inteiro) e exiba-a na tela no formato textual por extenso. Exemplo: Data: 01/01/2000, imprimir: 1 de janeiro de 2000.
19. Faça um função para verificar se um número é positivo ou negativo. Sendo que o valor de retorno será 1 se positivo, -1 se negativo e 0 se for igual a zero.
20. Faça uma função para verificar se um número é um quadrado perfeito. Ex: 1, 4, 9...
21. Faça uma função que receba dois números e retorne qual deles é o maior.
22. Elabore uma função que receba três notas de um aluno como parâmetros e uma letra. Se a letra for A, a função deverá calcular a média aritmética das notas do aluno; se for P, deverá calcular a média ponderada, com os pesos 5, 3, 2.
23. Faça um programa que escreva na tela, de 1 até 100 usando o while.
24. Faça um algoritmo utilizando o comando while que mostra uma contagem regressiva na tela, iniciando em 10 e terminando em 0. Mostrar uma mensagem "FIM!" após a contagem.
25. Faça um programa que leia 10 inteiros positivos, ignorando não positivos, e imprima sua média.
26. Escreva um programa que leia 10 números e escreva o menor valor lido e o maior valor lido.
27. Escreva um algoritmo que leia um conjunto de n números e mostre qual foi o menor e o maior valor fornecido.
28. Faça um algoritmo que leia um número inteiro e imprima seus divisores.
29. Faça um programa que receba um valor inteiro  $n \geq 0$  e imprima se esse número é primo ou não.
30. Escreva um algoritmo que leia certa quantidade de números inteiros e imprima o maior deles e quantas vezes o maior número foi lido. A quantidade de números a serem lidos deve ser fornecida pelo usuário.
31. Faça um programa que receba dois números. Calcule e mostre:
  - A soma dos números pares desse intervalo de números (intervalo incluindo os números dados);
  - A multiplicação dos números ímpares desse intervalo (intervalo incluindo os números dados)
32. Faça um algoritmo que seja capaz de obter o quociente inteiro da divisão de dois números fornecidos, sem utilizar a operação de divisão (/), nem a operação de divisão inteira (//) e nem a operação de resto da divisão inteira (%).
33. Dado um número n inteiro e positivo, dizemos que n é perfeito se n for igual à soma de seus divisores positivos diferentes de n. Construa um programa que verifique se um dado número é perfeito. Ex: 6 é perfeito, pois  $1 + 2 + 3 = 6$ .
34. Escreva um programa que gere todos os números entre 1000 – 1999 e mostra aqueles que divididos por 11 dão resto 5.
35. O número 3025 possui uma característica interessante, sendo a seguinte:  $30 + 25 = 55$  e  $55^2 = 3025$ . Elaborar um algoritmo que verifique todos os números de quatro algarismos que apresentem essa propriedade.
36. Faça uma função que retorne o maior fator primo de um número.

37. Faça uma função chamada de simplificada que receba como parâmetro o numerador e o denominador de uma fração. Esta função deve simplificar a fração recebida dividindo o numerador e denominador pelo maior fator possível. Por exemplo, a fração 36/60 simplificada para 3/5 dividindo o numerador e denominador por 12.
38. Crie uma função que diga se duas strings são iguais ou não, analisando caractere a caractere.
39. Escreva um programa que leia duas palavras e diga qual delas vem primeiro na ordem alfabética. Não use nenhuma funcionalidade do python que já faça isso. Dica: 'a' é menor do que 'b'.
40. Faça um programa que conte o número de 1's que aparecem em uma string. Exemplo: 0011001 -> 3. Não use nenhuma funcionalidade do python que já faça isso.
41. Escreva um programa que substitui as ocorrências de um caractere 0 em uma string pelo caractere 1. Não use nenhuma funcionalidade do python que já faça isso.
42. Ler uma frase e contar quantos caracteres são brancos. Não use nenhuma funcionalidade do python que já faça isso.
43. Escreva um programa que leia o nome e a idade de várias pessoas. Seu programa deve terminar quando uma idade negativa for digitada. Ao terminar, seu programa deve escrever o nome e a idade das pessoas mais jovens e mais velhas.
44. Faça um programa que receba uma frase e imprima-a de maneira invertida, trocando as letras A (maiúsculas ou minúsculas) por \*. Não use nenhuma funcionalidade do python que já faça isso.
45. O código de César é uma das mais simples e conhecidas técnicas de criptografia. É um tipo de substituição na qual cada letra do texto é substituída por outra, que se apresenta no alfabeto abaixo dela um número fixo de vezes. Por exemplo, com uma troca de três posições, 'A' seria substituído por 'D', 'B' se tornaria 'E', e assim por diante. Implemente uma função que faça uso desse Código de César, entre com uma string e a quantidade de posições e retorne a string codificada. Exemplo:

String: A LIGEIRA RAPOSA MARROM SALTOU SOBRE O CACHORRO  
CANSADO

Nova string: D OLJHLUD UDSRVD PDUURP VDOWRX VREUH R FDFKRUUR  
FDQVDGR

Observação: caso a letra codificada passe da letra Z, ela deve voltar para o início do alfabeto.

46. Faça uma função que, dada uma string, diga se ela é um palíndromo ou não. Lembrando que um palíndromo é uma palavra que tenha a propriedade de poder ser lida tanto da direita para a esquerda como da esquerda para a direita. Exemplos:

ovo

arara

Socorram-me, subi no onibus em Marrocos.

Anotaram a data da maratona