

essencial

A - Somar algarismos pares

Construa um programa que calcule a soma dos algarismos pares de um número inteiro positivo introduzido pelo utilizador.

Por exemplo, se a entrada for o número 36781 a saída será 14 (8+6)

Exemplo:

Entrada	Saída
36781	14

essencial

B - Somar algarismos pares (v2)

Construa um programa que calcule a soma dos algarismos pares de um número inteiro positivo introduzido pelo utilizador.

Por exemplo, se a entrada for o número 36781 a saída será 14 (8+6)

O algoritmo deve ser aplicado a uma sequência de números inteiros terminada por um número não positivo.

Exemplo:

Entrada	Saída
36781	14
4567	10
10003	0
-1	

essencial

C - Produto dos algarismos ímpares

Construa um programa que calcule o produto dos algarismos ímpares de um número inteiro positivo introduzido pelo utilizador.

Por exemplo, se a entrada for o número 56983 a saída será 135 (3*9*5).

Se o número não contiver algarismos ímpares deve ser visualizada a mensagem "nao ha algarismos ímpares".

Exemplo:

Entrada	Saída
56983	135

essencial

D - Produto dos algarismos impares (v2)

Construa um programa que calcule o produto dos algarismos ímpares de um número inteiro positivo introduzido pelo utilizador.

Por exemplo, se a entrada for o número 56983 a saída será 135 ($3 \cdot 9 \cdot 5$).

Se o número não contiver algarismos impares deve ser visualizada a mensagem "nao ha algarismos impares".

O algoritmo deve ser aplicado a uma sequência de números inteiros terminada por um número não positivo.

Exemplo:

Entrada	Saída
56983	135
321	3
1234567	105
-1	

essencial

E - Converter de base 8 para decimal

Construa um programa que recebe um número inteiro na base 8 e converte-o para o sistema decimal.

O algoritmo deve ser aplicado a uma sequência de números inteiros terminada por um número não positivo.

Exemplo:

Entrada	Saída
12345	5349
5000	2560
111	73
-1	

essencial

F - Capicua

Construa um programa que recebe um número inteiro positivo e verifique se é ou não uma capicua.

O resultado deve ser a mensagem "capicua" ou "nao capicua".

Exemplo1:

Entrada	Saída
12345	nao capicua

Exemplo2:

Entrada	Saída
5005	capicua

essencial

G - Divisores de um número

Construa um programa que dado um número inteiro positivo escreve todos os seus divisores e, no final, a quantidade de divisores entre parêntesis.
Todos os valores devem ser escritos em linhas separadas.

Exemplo:

Entrada	Saída
30	1 2 3 5 6 10 15 30 (8)

essencial

H - Número primo

Construa um programa que dado um número inteiro positivo verifica se é ou não um número primo.
Um número é primo se for inteiro e se só for divisível, por ele próprio e por 1.

O resultado deve ser a mensagem "primo" ou "nao e primo".

Exemplo1:

Entrada	Saída
27	nao e primo

Exemplo2:

Entrada	Saída
31	primo

I - Mínimo múltiplo comum

Construa um programa que dados dois números inteiros positivos calcula o mínimo múltiplo comum.

Exemplo1:

Entrada	Saída
12 6	12

Exemplo2:

Entrada	Saída
5 4	20

J - Percentagem de dígitos divisores do próprio número

Construa um programa que leia uma sequência de N números inteiros positivos, sendo N definido pelo utilizador.

Se o valor de N for negativo o algoritmo termina de imediato.

O algoritmo deve apresentar numa linha separada, para cada um dos números lidos, a percentagem dos dígitos que são divisores do próprio número (usando 2 casas decimais). O dígito zero (0) não deve ser considerado como potencial divisor mas deve ser contabilizado como dígito.

No final deve mostrar a maior dessas percentagens entre prêntesis.

Exemplo1:

Entrada	Saída
3 123 6 200	66.67% 100.00% 33.33% (100.00%)

Exemplo2:

Entrada	Saída
-6	

K - Números primos até um limite

Construa um programa que determine e visualize os números primos até um determinado valor N inserido pelo utilizador.

Um número é primo se for inteiro, maior que 1 e se só for divisível, por ele próprio e por 1.

Cada número deve aparecer numa linha separada..

Exemplo:

Entrada	Saída
16	2 3 5 7 11 13

essencial

L - Números perfeitos

Construa um programa que determine e visualize os N primeiros números perfeitos.

Um número é perfeito se for natural e for igual à soma de todos os seus divisores (excluindo o próprio número).

Cada número deve aparecer numa linha separada.

Exemplo:

Entrada	Saída
2	6 28

essencial

M - Sequência de Fibonacci

Construa um programa para mostrar os N primeiros termos da sucessão de Fibonacci.

Nesta sucessão, o primeiro termo é zero (0), o segundo termo é um (1) e qualquer um dos outros termos é igual à soma dos dois termos anteriores.

Cada número deve aparecer numa linha separada.

Exemplo:

Entrada	Saída
5	0 1

1
2
3

essencial

X - Relógio

Construa um programa em que dado um valor inteiro, representativo de um código de um relógio, indique a marca do mesmo.

A tabela seguinte indica a correspondência entre o dígito e a marca.

Código	Marca
1	Tag Heuer
2	Rolex
3	Omega
4	Cartier
5	Bvlgari
6	Raymond Weil
Outro	Marca invalida

Exemplo1:

Entrada	Saída
5	Bvlgari

Exemplo2:

Entrada	Saída
3	Omega

Exemplo3:

Entrada	Saída
8	Marca invalida

essencial

Y - Divisores múltiplos de 3

Construa um programa que leia um número inteiro positivo e apresente todos os seus divisores que sejam múltiplos de 3.

Deverá visualizar um divisor por linha.

Caso não existam divisores deverá ser visualizada a mensagem "sem divisores multiplos de 3".

Exemplo1:

Entrada	Saída
30	3 6 15

Exemplo2:

Entrada	Saída
29	sem divisores multiplos de 3

essencial

Z - Percentagem de algarismos pares e maior ímpar

Construa um programa em que dado um número inteiro positivo, determine a percentagem de algarismos pares e o maior algarismo ímpar.

Mostre a percentagem com 2 casas decimais e os resultados em linhas separadas.

Se não existirem algarismos ímpares deverá ser enviada a mensagem "nao ha algarismos impares".

Exemplo1:

Entrada	Saída
12345	40.00% 5

Exemplo2:

Entrada	Saída
2004	100.00% nao ha algarismos impares