

LISTA DE EXERCÍCIOS DE PROGRAMAÇÃO

- 1) Escreva um algoritmo que leia o número de litros vendidos, o tipo de combustível (codificado da seguinte forma: 1 – álcool; 2 – gasolina), calcule e imprima o valor a ser pago pelo cliente, sabendo que o preço da gasolina é de R\$ 3,15 e o litro do álcool é de R\$ 2,83.
- 2) Desenvolva um programa onde o usuário informe a idade e se tem alguma doença crônica. Caso tenha alguma doença crônica, deve-se informar também o número de doenças. O valor da mensalidade do plano de saúde sofrerá um acréscimo conforme a tabela abaixo, e no final o programa tem que informar qual o valor total a ser pago na mensalidade.

Idade	Taxa Básica	Add por Doença
0 – 18 anos	83,15	0%
19 – 33 anos	133,88	5%
33 – 44 anos	203,73	10%
45 – 58 anos	312,44	15%
Acima de 58 anos	498,53	30%

- 3) Faça um algoritmo que leia a idade de um atleta e escreva em que categoria ele se enquadra, seguindo o quadro abaixo:

Faixa Etária	Categoria
De 5 a 10 anos	Infantil
De 11 a 17 anos	Juvenil
De 18 a 30 anos	Profissional
Acima de 30 anos	Sênior

- 4) Para participar da categoria OURO do 1º. Campeonato Mundial de bolinha de Gude o jogador deve pesar entre 70 Kg (inclusive) e 80 kg (inclusive) e medir 1,75 m (inclusive) e 1,90 m (inclusive). Leia a altura e o peso de um jogador e determine se o jogador está apto a participar do campeonato escrevendo uma das seguintes mensagens, conforme cada situação:
 - “RECUSADO POR ALTURA” – somente se a altura do jogador for inválida.
 - “RECUDADO POR PESO”- somente se o peso do jogador for inválido.
 - “TOTALMENTE RECUSADO”- se a altura e o peso do jogador forem inválidos.
 - “ACEITO”- se a altura e o peso do jogador estiverem dentro da faixa especificada.
- 5) A Federação de Futebol contratou você para escrever um programa para fazer uma estatística de vários jogos (Flamengo e Vasco). Escreva um algoritmo para ler o número de gols marcados pelo Flamengo e o número de gols marcados pelo Vasco em um jogo,

imprimindo o nome do vencedor ou a palavra EMPATE. Logo após imprima a mensagem “Novo jogo (1 – Sim; 2 – Não)?”. Se a resposta for 1 o algoritmo deve ser executado novamente solicitando o número de gols marcados em uma nova partida, caso contrário deve ser encerrado e impresso:

- a. Quantos jogos fizeram parte da estatística.
 - b. O número de vitórias do Flamengo.
 - c. O número de vitórias do Vasco.
 - d. O número de empates.
 - e. Uma mensagem indicando qual time venceu o maior número de jogos ou NÃO HOUVE VENCEDOR.
- 6) Escreva um programa para ler um valor A e um valor N. Imprimir a soma do N números a partir de A (inclusive). Caso N seja negativo ou zero, deverá ser lido um novo número para N (só para N).

Valores para testes:

A = 3; N = 2; Soma = 7 (3+4)

A = 4; N = 5; Soma = 30 (4+5+6+7+8)

A = -2; N = 4; Soma = -2 (-2+ -1 + 0 + 1)

- 7) Faça um programa para exibir e calcular a soma dos N primeiros termos da sequência de Fibonacci. Esta sequência começa com os termos 1 e 1 e, a partir do terceiro termo, os termos são calculados pela soma dos dois termos anteriores: 1, 1, 2, 3, 5, 8, 13, 21, 34, ...
- 8) Faça um programa para calcular o Máximo Divisor Comum entre 2 números. Observe as seguintes propriedades do MDC:
- $MDC(x, y) = MDC(x - y, y)$, se $x > y$;
- $MDC(x, y) = MDC(y, x)$;
- $MDC(x, x) = x$.
- Exemplo: $MDC(3,5)=MDC(5,3)=MDC(2,3)=MDC(3,2)=MDC(1,2)=MDC(2,1)=MDC(1,1)=1$
- 9) Elabora uma função que permita somar os elementos de um vetor de inteiros.

- 10) Utilizando switch case, faça um programa que contenha um menu com as seguintes opções:

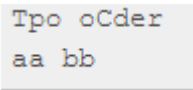
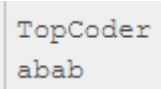
- a. Ler uma string S1 (tamanho máximo 20 caracteres);
- b. Imprimir o tamanho da string S1;
- c. Comparar a string S1 com uma nova string S2 fornecida pelo usuário e imprimir o resultado da comparação;
- d. Concatenar a string S1 com uma nova string S2 e imprimir na tela o resultado da concatenação;
- e. Imprimir a string S1 de forma reversa;
- f. Contar quantas vezes um dado caractere aparece na string S1. Esse caractere desse ser informado pelo usuário;
- g. Retornar uma substring da string S1. Para isso o usuário deve informar a partir de qual posição deve ser criada a substring e qual é o tamanho da substring.
- h. Substituir a primeira ocorrência do caractere C1 da string S1 pelo caractere C2. Os caracteres C1 e C2 serão lidos pelo usuário;

- 11) Faça um programa que receba do usuário uma string. O programa imprime a string sem suas vogais.
- 12) Fazer um programa em C para ler um vetor de inteiros positivos de 50 posições. Imprimir a quantidade de números primos presentes nele.
- 13) Leia dois vetores inteiros x e y , cada um com 5 elementos (assuma que o usuário não informa elementos repetidos). Utilizando a estrutura switch case, faça:

Código	Ação
S	Soma entre cada elemento de x com o elemento da mesma posição em y .
P	Produto entre cada elemento de x com o elemento da mesma posição em y .
M	Subtração entre cada elemento de x com o elemento da mesma posição em y .
I	Interseção entre x e y .
D	Elementos diferentes entre x e y .

O programa só deve parar de executar ao ser digitado a letra Z.

- 14) Faça um programa para ler 10 números DIFERENTES a serem armazenados em um vetor. Os dados deverão ser armazenados no vetor na ordem que forem sendo lidos, sendo que caso o usuário digite um número que já foi digitado anteriormente, o programa deveria pedir para ele digitar outro número. Note que cada valor digitado pelo usuário deve ser pesquisado no vetor, verificando se ele existe entre os números que já foram fornecidos. Exibir na tela o vetor final que foi digitado.
- 15) Faça um vetor de tamanho 50 preenchido com o seguinte valor: $(i + 5 * i) \% (i + 1)$, sendo i a posição do elemento no vetor. Em seguida imprima o vetor na tela.
- 16) Implemente um programa Combiner que receba dois Strings como parâmetros e combine-os, alternando letras, começando com a primeira letra da primeira String, seguida pela primeira letra da segunda String, depois pela segunda letra da primeira String, etc. As letras restantes da String mais longa são então anexadas ao final da string de combinação e esta string de combinação é retornada. Cada string conterá entre 1 e 50 caracteres.

Exemplo: Entrada  Saída 

- 17) Júlio César usou um sistema de criptografia, agora conhecido como Cifra de César, que deslocava cada letra 2 lugares para frente pelo alfabeto (por exemplo, 'A' muda para 'C', 'R' muda para 'T', etc.). No final do alfabeto nós damos a volta, isso é, 'Y' muda para 'A'. Podemos, é claro, tentar mudar qualquer número.

Dado um número e uma string já codificada utilizando a cifra de César, decodifique-a e imprima a sentença original.

Exemplos:

Entrada:

```
6
VQREQFGT
2
ABCDEFGHIJKLMNOPQRSTUVWXYZ
10
TOPCODER
0
ZWBGLZ
25
DBNPCBQ
1
LIPPSASVPH
4
```

Saída:

```
TOPCODER
QRSTUVWXYZABCDEFGHIJKLMNOP
TOPCODER
AXCHMA
CAMOBAP
HELLOWORLD
```