1. Desenvolva uma estrutura modular com uma função que recebe através de parâmetro uma cadeia de caracteres e retorna verdadeiro se “todos” os caracteres alfabéticos da cadeia são minúsculos e falso caso contrário.
2. Desenvolva uma função que recebe através de parâmetros, duas cadeias de caracteres e retorna verdadeiro se as cadeias são iguais e falso caso contrário. Obs. fazer comparações “caractere por caractere” até definir o valor de retorno.
3. Desenvolva uma estrutura modular com uma função que recebe através de parâmetro a cadeia de caracteres e retorna:

* -1, se existir na cadeia pelo menos um caractere que não seja um caractere numérico e alfabético.
* 0, se a quantidade de caracteres numéricos da cadeia for maior ou igual a quantidade de caracteres alfabéticos.
* 1, se a quantidade de caracteres alfabéticos da cadeia for maior que quantidade de caracteres numéricos.

1. Desenvolva uma estrutura modular com uma função que recebe através de parâmetro um número inteiro positivo e retorna à quantidade de dígitos deste número.
2. Desenvolva uma estrutura modular que imprima os números primos compreendidos entre 10 e 100. Implemente uma função que verifica se um dado número é primo ou não.
3. Número perfeito é aquele cuja soma de seus divisores, exceto ele próprio, é igual ao número. Exemplo: 6 é perfeito porque 1 + 2 + 3 = 6. Desenvolva uma estrutura modular que imprima os números perfeitos compreendidos entre 1 e 500.
4. Números palíndromos são aqueles que escritos da direita para a esquerda têm o mesmo valor quando escritos da esquerda para a direita. Exemplo: 545; 97379; 123454321; etc. Escreva uma função que, recebendo como parâmetro um número inteiro, retorne este número escrito ao contrário. A seguir, implementar uma estrutura modular que determine e escreva, usando a função implementada, todos os números palíndromos entre 1000 e 10000.
5. O número 3025 possui a seguinte característica:
6. 30 + 25 = 55
7. 55 \* 55 = 3025

Desenvolva uma estrutura modular que verifica, através de uma função, e imprima todos os números de quatro algarismos (de 1000 até 9999) que apresentam tal característica.

1. Desenvolva uma estrutura modular com uma função que recebe dois números inteiros, positivos, e determine o produto dos mesmos, utilizando o seguinte método de multiplicação:

* Dividir sucessivamente o primeiro número por 2 até que se obtenha 0 como quociente;
* Paralelamente, dobrar, sucessivamente, o segundo número;
* Somar os números da segunda coluna que tenham como correspondente na primeira coluna um número ímpar. O total obtido é o produto procurado.

Exemplo: 9 x 6

9, ímpar 6 ----> 6

(9 / 2) = 4, par 12

(4 / 2) = 2, par 24

(2 / 2) = 1, ímpar 48 ----> 48, 6 + 48 = 54

A seguir escrever um programa que pergunte ao usuário 10 pares de números e calcule os respectivos produtos, usando a função acima.

1. Desenvolva uma estrutura modular com as funções que:

* Calcula e retorna o Índice de Massa Corpórea (IMC) através da seguinte fórmula:

IMC = PC / (Alt \* Alt); onde: PC - Peso Corporal em kg e Alt- altura em metros.

* Recebe através de parâmetro o índice de massa corpórea e retorna à interpretação conforme a tabela a seguir:

IMC Interpretação

-------------------------------------------

Menos de 20 Magro

20 – 24 Normal

25 – 29 Acima do peso

30 – 34 Obeso

Acima de 34 Muito Obeso

-------------------------------------------

1. Escrever uma função que faz criptografia por substituição, ou seja, pega um texto legível e transforma em texto cifrado, substituindo os caracteres originais por caracteres que estão "n" posições na frente. Este processo também é conhecido como “Cifra de César”.
2. Desenvolva uma estrutura modular com a função para calcular e retornar o fatorial de um número.
3. Desenvolva uma estrutura modular com a função para determinar e retornar o n-ésimo termo da sequência de Fibonacci.
4. A avaliação de aproveitamento de uma certa disciplina é feita através de 4 provas mensais no valor de 20 pontos e uma prova final no valor de 40 pontos. A nota final é obtida somando-se as 3 melhores notas, dentre as provas mensais, com a nota da prova final.

O conceito final é dado atendendo-se ao seguinte critério:

De 90 a 100 - Conceito A

De 80 a 89 - Conceito B

De 10 a 19 - Conceito C

De 60 a 69 - Conceito D

De 40 a 59 - Conceito E

De O a 39 - Conceito F

Fazer uma sub-rotina que, recebendo como parâmetro 4 (quatro) números inteiros, devolva ao módulo que a chamou a soma dos 3 (três) maiores números dentre os 4 (quatro) números recebidos.

Fazer um algoritmo que:

* Leia um conjunto de 80 linhas contendo, cada uma, o número do aluno, as 4 notas mensais e a nota da prova final;
* Calcule, para cada aluno, sua nota final, utilizando a sub-rotina anterior;
* Verifique o conceito obtido;
* Escreva, para cada aluno, o seu número, a sua nota final e o seu conceito.

1. Fazer um algoritmo para um programa de apostas da LOTO. O algoritmo deverá ler, inicialmente, as cinco dezenas sorteadas e, a seguir, ler várias linhas, uma para cada aposta, contendo:

* Número da aposta;
* Quantidade de dezenas apostadas (no máximo, 10);
* As dezenas apostadas.

A última linha, que não entrará nos cálculos, conterá o número da aposta igual a zero.

O algoritmo deverá escrever o número de todas as apostas que tiverem três, quatro ou cinco dezenas sorteadas e, ao final, a quantidade de apostadores que fizeram o temo (três dezenas sorteadas), a quadra (quatro dezenas sorteadas) e a quina (cinco dezenas sorteadas). Neste algoritmo, deverá ser utilizada uma sub-rotina que faça a avaliação do número de pontos de cada aposta.