

Lista de atividades 10: funções

Codifique algoritmos abaixo na linguagem C.

01) Fazer uma função *calculaRaiz* que recebe como parâmetro os coeficientes de uma equação quadrática (coeficientes a , b e c) e retorna três informações que são as duas raízes (x' e x'') e uma informação se as raízes são reais.

Utilize essa função para mostrar as raízes de uma equação quadrática com os coeficientes lidos a partir da entrada padrão. Caso as raízes não sejam reais mostre a mensagem “*Raizes nao sao numeros reais*”.

02) Faça a função *potência* que recebe os inteiros x e n como parâmetros e mostra o resultado x^n . Faça um algoritmo que lê dois números inteiros x e n , e utilize a função *potência* para mostrar o resultado da potência x^n .

03) Faça uma função que recebe os parâmetros peso, altura e retorna o IMC (peso dividido pelo quadrado da altura). A seguir, faça um programa que lê peso e altura de uma pessoa e utiliza a função para calcular o IMC. Se o IMC estiver acima de 25 escreva a mensagem: “*acima do peso*”

04) Faça a função *menor2* que recebe 2 números e retorna o menor valor. Faça um algoritmo para ler 2 números e utilizar a função *menor2*.

05) Faça a função *menor3* que recebe 3 números e retorna o menor valor, reutilize a função *menor2* nessa função. Faça um programa para ler 3 números e utilizar a função *menor3*.

06) Faça a função *primo* que recebe como parâmetro um número n e mostra todos os números primos de 1 até n .
Note que um número primo é aquele que é divisível somente por 1 e por ele mesmo. A seguir, faça a um algoritmo para ler o número n e utilizar a função *primo*.

07) Faça a função *mdc* que recebe como parâmetros dois números inteiros e retorna o máximo divisor comum (MDC) entre esses dois números mostra todos os números primos de 1 até n .
A seguir, faça um programa para ler três números inteiros e calcular o MDC dentre esses números utilizando a função *mdc*.

08) Faça a função *distancia* que retorna a distância euclidiana entre dois pontos $p1$ e $p2$. Para isso a função deve utilizar o tipo abstrato de dados (TAD) *ponto* composto pelas coordenadas x e y . Faça um algoritmo que lê os valores para os dois pontos e utiliza a função *distancia* para mostrar a distância entre os pontos.
Dica: utilize o algoritmo desenvolvido na questão 4 da lista de atividades anterior.