

PROGRAMAÇÃO ESTRUTURADA (MODULAR)		PROGRAMAÇÃO ESTRUTURADA (MODULAR)
O que seria a Programação Estruturada ou Modular ?		Consiste na divisão de grandes tarefas de computação em partes menores, a fim de utilizar seus resultados parciais para compor o resultado final desejado. Dessa forma, há uma diminuição da extensão dos programas, de forma que qualquer alteração poderá ser feita mais rapidamente, caso seja necessária.
O que é a técnica de Modularização ?		A modularização é uma técnica utilizada para desenvolver algoritmos, na qual se divide o problema em pequenas partes denominadas módulos, sendo estes também conhecidos também como sub-rotinas, sub-programas ou sub-algoritmos.
Quais as vantagens de se usar a técnica de Modularização?		A principal vantagem da modularização é possibilitar o reaproveitamento de código, já que podemos utilizar um módulo várias vezes, eliminando assim a necessidade de escrever o mesmo código em situações repetitivas. As vantagens da programação estruturada são: - Deixar o programa mais legível e fácil de entender; - Encontrar e corrigir erros com mais facilidade; - Facilitar os testes, diminuindo a probabilidade de erros; - Reduzir o tempo e custo da programação; - Possibilitar o reaproveitamento de partes do programa (módulos);
Quais são os 2 tipos de Módulos básicos que existem?		Basicamente, há 2 tipos de módulos: Procedimentos e Funções . Entre esses dois tipos de módulos existem algumas diferenças, mas o conceito é mesmo para ambas. <i>O importante no uso prático desses dois tipos de módulo é distinguir as diferenças entre eles e como utilizá-los no momento mais adequado.</i>
Como se dá o Módulo Procedimento ?		Um procedimento é um bloco de programa contendo início e fim, identificado por um nome, por meio do qual será referenciado em qualquer parte do programa principal ou em outro procedimento. A característica principal de um procedimento é que não há retorno de valor ou variável, pois, no momento da execução do programa, somente as instruções definidas dentro do escopo do procedimento são executadas. Quando um procedimento é chamado por um programa, ele é executado até o seu término e a execução do programa volta exatamente para a primeira linha de instrução, após a linha que fez a chamada do procedimento.
- reservada para a questão acima -		A forma geral para se definir um procedimento é a seguinte: <i>void nome-do-procedimento (lista de parâmetros){ corpo do procedimento (instruções) }</i>
Ilustre como podemos implementar um Procedimento...		Temos um Procedimento de Entrada para uma calculadora, que quando for chamado solicitará a entrada ao usuário das variáveis A e B... <i>void Entrada () { printf("\nDigite o primeiro numero: "); scanf("%f",&A); printf("\nDigite o segundo numero: "); scanf("%f",&B); }</i>
Como se dá o Módulo Função ?		Uma função também é um bloco de programa, como são os procedimentos, contendo início e fim e identificada por um nome, pelo qual também será referenciada em qualquer parte do programa principal. A principal diferença entre um procedimento e uma função está no fato de uma função retornar um determinado valor, sendo que o tipo de dado (int, float, char) de tal valor deve ser definido na criação da função.
- reservada para a questão acima -		A forma geral para se definir uma função é a seguinte: <i>tipo-de-dado nome-da-função (lista de parâmetros){ corpo da função (instruções) return (variável/valor); }</i>
Ilustre como podemos implementar uma Função...		A seguir, apresentamos a definição do Módulo Adição citado no exemplo da calculadora. Quando esta função for chamada e executada, será feito o cálculo de adição das variáveis A e B, o qual será atribuído à variável R, que será o retorno da função. <i>float Adicao () { R = A + B; return (R); }</i>

