

1.TAREFA PARA FINAL DE SEMANA

Há dois tipos básicos de instruções que podem ser declaradas pelo programador: procedimentos e funções. A principal diferença entre elas é que a função retorna um valor. Poder declarar instruções, estendendo a linguagem, certamente é o recurso mais valioso das linguagens de programação, por proporcionar reusabilidade e portanto produtividade. É conveniente também saber como armazená-las em bibliotecas, para que elas sejam *facilmente* reutilizáveis.

Mas lembre-se: uma instrução só será reutilizável se puder ser usada sem qualquer outra declaração, isto é, é preciso saber apenas o seu nome e seus parâmetros. Também não será conveniente que a instrução mostre resultados na tela, a menos que isso faça parte da sua finalidade. Por exemplo, se a finalidade for "calcular a média", a instrução retorna o valor calculado mas não mostra na tela. Se a finalidade for "Mostrar uma mensagem na tela", então ela deve fazer isso e não efetuar cálculos de valores que não estejam relacionados a isso.

Caso a instrução especificada em algum exercício deste capítulo já exista na linguagem em que estiver programando, é claro que você tem pelo menos duas opções: usar a disponível ou desenvolver a sua própria, para treinar. Neste caso, você pode comparar os resultados da sua com a da linguagem, para validar a instrução criada.

Nas especificações que se seguem, se for pedido "declare uma instrução" ou semelhante, você deverá fazer um pequeno programa para testá-la.

MATEMÁTICA

Número par - Fazer um procedimento que retorna Verdadeiro ou Falso conforme um número seja par ou não. Se necessário, convencie 0 e 1, "S" e "N" ou outra representação de Falso e Verdadeiro.

Numeração de 1 a 100 - Elaborar um procedimento que mostra os números de 1 a 100.

Numeração de N1 a N2 - Declarar um procedimento semelhante ao acima, mas que recebe como parâmetros os valores inicial e final.

Números pares - Escrever um procedimento que recebe dois números e mostra na tela os números pares situados entre os dois, inclusive. Testar com um programa que lê os números inicial e final e, se este maior que o inicial, chama o procedimento.

Equação do segundo grau - Implementar uma instrução que recebe os coeficientes a, b e c de uma equação do segundo grau e retorna suas raízes. Resolva: como fazer quando a equação resultante não tiver raízes reais?

Aprovação - Escrever uma função que recebe uma nota de 0 a 10 e retorna verdadeiro ou falso (ou outros valores convencionados como tal) se o aluno foi aprovado, isto é, se tirou 7 ou mais.

Maior e menor com menu - Faça duas funções: uma que recebe dois números e retorna o maior e outra que recebe o mesmo mas retorna o menor. Implementar um programa com um menu de 4 opções: ler dois números, testar uma e outra função e terminar.

Maior de 2 - Escrever uma função que recebe dois números quaisquer e retorna o maior.

Maior de 3 - Escrever uma função que recebe 3 números e retorna o maior

Entre 0 e 100 - Implementar uma função que recebe um número qualquer e retorna Verdadeiro se o número está entre 0 e 100, caso contrário retorna Falso.

Média de 3 - Declarar uma função que calcula a média aritmética de 3 números.

Exponenciação inteira - Escrever uma função que calcula um inteiro elevado a outro inteiro, usando multiplicação.

Exponenciação real - Escrever uma função que calcula um número real elevado a outro real, usando multiplicação. Se houver tal função na linguagem em que estiver programando, compare seus resultados para vários tipos de valores.

Juros compostos - Sendo C o capital, n o prazo e i a taxa de juros, o valor futuro FV de uma aplicação financeira é calculado por:

$$FV = C(1 + i)^n$$

Elabore um algoritmo/programa com duas opções: uma que, dados capital, prazo e taxa, calcule o valor futuro, e outra que, dados prazo, taxa e valor futuro, calcule o capital necessário.

Série - Elaborar programa que calcule, usando função, o valor da série abaixo para N termos, sendo N lido:

$$S = 1 - \frac{1}{3^2} + \frac{1}{5^2} - \frac{1}{7^2} + \frac{1}{9^2} - \dots$$

Bissexto - Um ano é bissexto se for divisível por 4 exceto os séculos, que são bissextos se forem múltiplos de 400. Implementar uma função que recebe o número de um ano e retorna Verdadeiro se o ano for bissexto ou Falso caso contrário.

Fatorial - Faça uma função que recebe como parâmetro um inteiro e retorna seu fatorial. O que você acha que poderia fazer para o caso em que o fatorial, se calculado, vai estourar a capacidade do tipo de dado adotado?

Números primos - Um número é dito ser primo quando é divisível somente por si e pela unidade. Faça um programa que verifica, através de uma função, se um número é ou não primo. [Dica: divida o número N por todos os números de 2 a $N - 1$. Se o resto da divisão de N por algum dos números der zero, ele não é primo]

Arco-tangente - O valor do arco-tangente pode ser calculado através da fórmula abaixo, válida quando $x^2 \leq 1$:

$$x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \frac{x^9}{9} - \dots$$

Elabore um algoritmo/programa que lê o valor de x e a quantidade de fatores, e compara o valor encontrado com o calculado pela respectiva instrução da linguagem. Teste para quantidades variadas de fatores.

CARACTERES E CADEIAS

Menor cadeia - Escrever uma função que recebe duas cadeias de caracteres e retorna a menor em ordem alfabética.

Leitura de cadeia - Implementar um procedimento que recebe uma linha e uma coluna e lê uma variável caractere na respectiva posição da tela, retornando o valor lido através de um parâmetro por referência ou equivalente.

Cabeçalho - Criar procedimento que recebe e imprime uma linha de cabeçalho com número de página, no formato. A data é obtida do sistema e o número de página é um parâmetro:

RELATÓRIO DE CLIENTES

EMIÇÃO: DD/MM/AAAA

PÁG. 999

Dia, mês e ano - Implemente funções que recebem uma data no formato 'DD/MM/AAAA' e retornam dia, mês e ano, respectivamente.

Validação de data - Escrever uma função que verifica se uma data no formato 'DD/MM/AAAA' é válida. Além de verificar se o dia está entre 1 e 31 e o mês está entre 1 e 12, se o dia for 29/02 é chamada a função especificada no item 0 para verificar se o ano é bissexto.

Formatação de data - Elaborar um procedimento que recebe como parâmetros dia, mês, ano, uma linha e uma coluna da tela e uma letra que corresponde ao formato de data a ser apresentado. Se o formato = 'A', mostra DD/MM/AA; se 'B', mostra DD.MM.AAAA. Para testá-lo, faça um algoritmo/programa que busca a data do sistema, extrai dia, mês e ano e chama o procedimento.

Inversão de cadeia - Escrever uma função que recebe uma cadeia de caracteres e retorna-a invertida (lida de trás para a frente).

Palíndromos - Implementar um algoritmo/programa que verifica se uma frase é palíndroma (a mesma se lida normalmente ou de trás para a frente: "roma me tem amor", "socorram me subi no onibus em marrocos"). Use a função especificada acima e uma outra para retirar espaços de uma cadeia.

Ocorrências de subcadeia - Elabore uma instrução que identifica quantas vezes uma subcadeia ocorre em uma cadeia de caracteres (por exemplo, "na" ocorre duas vezes em "banana").

Maiúsculas - Implementar uma função que converte uma cadeia de caracteres para maiúsculas.

Minúsculas - Elaborar uma instrução que converte uma cadeia de caracteres para minúsculas. [Dica: subtraia 32 dos caracteres cujos códigos ASCII estão entre 65 e 90, inclusive].

Inserção de caractere - Elaborar uma função que insere um caractere recebido entre cada letra de uma cadeia.

Trim - Implementar uma função que retira os brancos finais de uma cadeia de caracteres (em certas linguagens disponível com o nome de "Trim")

Ajuste de tamanho - Implementar uma função que insere brancos no fim de uma cadeia até atingir um tamanho especificado.

Crítica de data - Escrever uma função que recebe uma data no formato 'DD/MM/AAAA' (dia/mês/ano), verifica se as barras estão na posição correta, se o dia está entre 1 e 31 (inclusive), se o mês está entre 1 e 12 e retorna um valor da seguinte forma:

- 0: data correta
- 1: dia inválido
- 2: mês inválido
- 3: dia e mês inválidos
- 4: formato inválido

Formatação de linha - Um programador está escrevendo um editor de textos, e precisa de uma função que recebe uma linha de texto e uma largura de linha, e insere espaços de forma a alinhar o texto à largura recebida, retornando a linha formatada. Implemente essa função. *[Dica: primeiro calcule a quantidade de espaços necessários; descubra a quantidade de intervalos entre palavras e calcule quantos espaços terá que inserir em cada intervalo; insira essa quantidade em cada intervalo, sendo que o último intervalo receberá os espaços extras restantes. Investigue outras possibilidades de distribuição]*