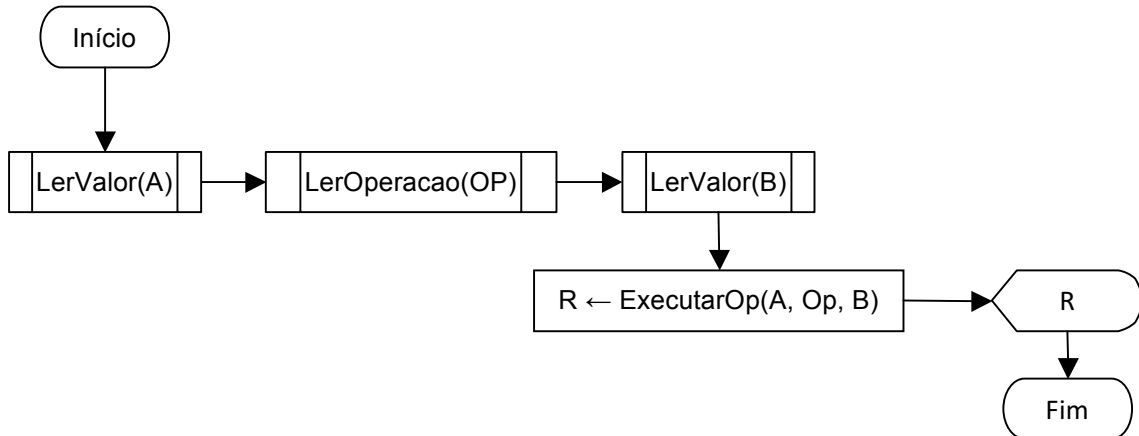


Lista de Exercícios 3 Subprogramação

Observações: Para todos os exercícios você deve: (i) caprichar na formatação dos dados e mensagens, ou seja, deixe as entradas e saídas com aparência profissional; (ii) comentar o código fonte produzido; e (iii) utilizar variáveis com identificadores auto-explicativos. Acostume-se a isso, pois será cobrado!

1. Modifique o programa escrito para o Exercício 6 da Lista 2 de modo que as conversões de segundos para minutos e minutos para horas seja feita por chamadas consecutivas de uma mesma função.
2. Escreva um programa FORTRAN que calcule a potencia de um número utilizando somente as operações de multiplicação e divisão. O usuário fornece o número real e o valor inteiro da potência. Após calculado, o resultado deverá ser exibido com cinco dígitos para a parte inteira do número e dois para a parte decimal. Encapsule a implementação de cálculo de potência em uma função.
3. Escreva um programa FORTRAN que pede para o usuário informar N valores inteiros. N deverá ser definido como PARAMETER no programa principal. Uma vez lidos os valores, o programa deverá reorganizá-los em ordem decrescente. Por fim, o programa deverá exibir o resultado da reorganização. Utilize subprogramação sempre que possível.
4. Escreva um programa FORTRAN que leia dois vetores de 10 posições, ordene cada vetor individualmente em ordem crescente, e combine os dois vetores gerando um novo vetor de 20 posições, de forma que esse novo vetor já seja criado ordenado. Ao final, mostre o vetor criado. Utilize subprogramação sempre que possível.
5. Escreva um programa FORTRAN que peça ao usuário que sejam informados dois valores inteiros e uma das quatro operações básicas da matemática, execute a

operação e mostre o resultado. O fluxo de processamento do programa deve estar estritamente de acordo com o seguinte fluxograma:



Note que `LerValor`, `LerOperacao` e `ExecutarOperacao` são sub-rotinas. É esperado que `A`, `B` e `R` sejam variáveis de tipo inteiro e `Op` uma variável do tipo caractere.

A sub-rotina `LerValor` deve pedir para o usuário que seja informado um número inteiro com até três dígitos. A sub-rotina receberá este valor dentro de uma variável do tipo caractere de tamanho 3. É obrigação de `LerValor` verificar se o texto informado pelo usuário corresponde a um valor numérico. Para isso, cada caractere deve ser inspecionado para verificar se esses correspondem aos caracteres `'0'` a `'9'`. Caso algum dos caracteres não seja numérico, a sub-rotina deve alertar o usuário e requisitar um novo valor. Caso seja válido, a sub-rotina deverá converter o texto para valor numérico inteiro e atribuir este valor ao argumento de `LerValor`. Dica 1: Utilize a função `ICHAR(X)` para converter um caractere (`X`) para seu correspondente inteiro na tabela ASCII e a função `CHAR(I)` para converter um valor inteiro (`I`) para seu caractere correspondente na tabela ASCII. Dica 2: Utilize a indexação de texto para obter um subconjunto de caracteres de uma variável do tipo texto. O uso desta indexação é ilustrado abaixo:

```
character*20 txt

txt = 'Texto para teste'

print *, txt(1:1)
print *, txt(7:7)
print *, txt(12:16)
```

onde o primeiro `print` imprime `'T'`, o segundo imprime `'p'` e o terceiro imprime `'teste'`.

A sub-rotina `LerOperacao` deve pedir para o usuário que seja informado uma das quatro operações básicas da matemática, identificadas pelos caracteres `'+'`, `'-'`, `'*'` e `'/'`. Caso o caractere informado pelo usuário seja diferente desses quatro, a sub-rotina deve alertar o usuário e requisitar a operação novamente. Caso seja válido, a sub-rotina deverá atribuir o caractere informado ao argumento de `LerOperacao`.

A função `ExecutarOp` deve aplicar sobre os argumentos inteiros `A` e `B` a operação informado em `Op` e retornar o resultado desta operação. Cada uma das operações deverá ser implementada em uma função a parte nomeadas, respectivamente, como: `Soma`, `Subtracao`, `Multiplicacao` e `Divisao`.