



Atenção:

- Faça corretamente os comentários e indentação de todos os programas.
 - Utilize constantes e definições de tipos sempre que possível.
 - Para os exercícios que pedem sub-rotinas, construa também um programa para usá-las.
 - Utilize a modularização a nível de unidade.
-

1. Considere uma **String** no formato **HH:MM:SS** para representar horas minutos e segundos. Construa as seguintes sub-rotinas:

Horas(Tempo: String): Integer Retorna a quantidade de horas do **String**;

Minutos(Tempo: String): Integer Retorna a quantidade de minutos do **String**;

Segundos(Tempo: String): Integer Retorna a quantidade de segundos do **String**;

Tempo(Horas, Minutos, Segundos: Integer): String Retorna uma **String** no formato **HH:MM:SS**.

2. Crie as sub-rotinas **PerguntaInteiro (TextoPergunta: String): Integer** e **PerguntaReal (TextoPergunta: String): Real**. Estas sub-rotinas devem receber um texto (**TextoPergunta**), imprimi-lo na tela e obter uma resposta textual do usuário. Entretanto, as sub-rotinas devem tratar digitações incorretas de números.
3. Crie a sub-rotina **TextoVezez(TextoPrincipal, TextoProcurado: String): Integer** que deve retornar quantas vezes o **TextoProcurado** aparece no **TextoPrincipal**.
4. Considere o seguinte formato de uma equação de segundo grau $Ax^2 + Bx + C = 0$. Crie a sub-rotina **TermosEquacaoGrau2(Equacao: String): TVetTermos** que retorne um vetor de 3 posições com os termos da equação.
5. Considere o tipo de vetor **TVetTexto = Array [1..MAX] of String**. Construa as seguintes sub-rotinas:

Ocorrencias(V: TVetTexto; Palavra: String): Integer Retorna quantas vezes uma **Palavra** aparece em **V**. Dica você pode usar a sub-rotina **TextoVezez()** para facilitar;

TotalCaracteres(V: TVetTexto): Integer Retorna quantos caracteres **V** possui, considerando todas as strings.