

E a lógica, como vai?

Existe a ilusão de que apenas conhecer uma boa linguagem credencia a bom desenvolvedor de sistema. Todavia, a Lógica de Programação tem papel fundamental na qualidade do programa.

Qual é a melhor linguagem?

"Qual é a linguagem do momento?" "Qual é o software mais usado no mercado hoje?" "Qual software você acha que devo aprender para estar atualizado em relação ao mercado?" "Qual é a Linguagem que mais tem futuro?"

É comum ouvirmos perguntas deste tipo de pessoas que desejam estar na vanguarda da Informática. Existe uma grande preocupação de atualizar-se em termos do melhor software de programação. Todavia, alguns, entusiasmados pelo software, esquecem daquilo que os move: a Lógica Computacional.

Balance Line?

Você já fez um "Balance Line"? Sabe o que é? Este termo (em desuso) significa fazer a consolidação de um ou mais arquivos seqüenciais, pré-ordenados, gerando um novo arquivo. Este era um problema típico a ser resolvido por um programador de Main Frame, quando o processamento Batch - em lotes - era largamente utilizado.

Programas como este exigiam do programador muito mais de sua lógica do que de seu conhecimento da linguagem de programação, ou seja, o maior esforço ficava por conta do raciocínio lógico para resolver o problema do que a codificação da solução em uma linguagem de programação.

O Retrabalho

Quando efetuamos um trabalho e este não está bem feito, naturalmente será preciso refazê-lo. Um programa mal construído, que não atinge o propósito a que foi concebido, está credenciado a ser refeito, mesmo que contenha requintes da linguagem. Programar "bonito", explorando bem os recursos que a linguagem proporciona não é garantia de que o programa desempenhará o papel esperado.

O que é um programa?

Uma das definições de programa é: "uma seqüência de instruções logicamente ordenadas a fim de solucionar um problema" .

Isto significa que, se desejamos resolver um problema, é preciso estabelecer quais passos precisamos efetuar e, depois disto, ordená-los de forma lógica a fim de que desempenhem o que deles se espera.

Elaborar Regras

Na verdade, o raciocínio Lógico utilizado para resolver um problema deve elaborar uma regra (ou várias regras) a ser seguida a fim de que o problema seja resolvido. A(s) regra(s) deve indicar por qual caminho a solução será alcançada.

E o Software?

Uma vez com a regra pronta, sabendo como chegar à solução, a próxima etapa é "traduzir" a regra em código, ou seja, programar a solução utilizando uma linguagem de programação.

E quem programa "direto"?

Você pode estar se perguntando: "mas e quem começa direto pela programação?"

Ora, isto é comum nos programadores com maior experiência, mas,

mentalmente, a solução foi arquitetada. Não há impeditivo a isto, mas é importante ter-se a solução antes da programação. Qualidade do Programa Para que um programa tenha qualidade, são necessários alguns itens:

- a) Requerimento: definição clara (e preferencialmente documentada) do que se espera que o programa realize;
- b) Estratégia de Solução: A elaboração da Regra para solução do problema a resolver, ou seja, a Lógica do Programa;
- c) Desenvolvimento Estruturado: desenvolver o algoritmo que solucionará o problema / efetuará o trabalho;
- d) Teste da Solução: Validar a solução construída (elaborar e efetuar casos de teste, verificando se o programa produz os resultados esperados).

Estratégia de Solução (Regra)

A solução lógica do problema, seja ela elaborada via um diagrama de blocos ou mesmo arquitetada mentalmente, constitui fator primordial para a boa qualidade do programa.

Exemplo de Lógica

Para ilustrarmos, imagine a seguinte necessidade: temos a variável de memória `mes`, numérica, com valores variando de 1 a 12, e precisamos encontrar as 3 primeiras letras do mês correspondente ("JAN", "FEV", "MAR", ...), armazenando-as na variável string `str`. Obviamente, há várias formas de resolver-se este problema, ou seja, não há uma forma lógica única de resolver-se. Porém, dependendo da solução lógica adotada, a programação será bem diferente. Naturalmente, privilegiando o reuso, deve ser construída uma função (ou rotina) que possa ser utilizada em várias oportunidades. Comentaremos 3 alternativas para construção desta função:

Primeira Forma - Ninho de IF's

```
IF mes = 1 then
  str := "JAN"
else
  if mes = 2 then
    str := "FEV"
  else
    if mes = 3 then
      str := "MAR" else ...
```

Obviamente este solução é longa e não é a mais adequada.

Segunda Forma - Vetor

Cria-se um vetor com 12 ocorrências, contendo as 3 primeiras letras dos 12 meses possíveis:

```
aMES [ 1 ] := "JAN"
```

```
aMES [ 2] := "FEV"  
aMES [ 3] := "MAR"
```

```
.  
. .  
.
```

```
aMES [12] := "DEZ"  
str := aMES[mes]
```

Esta é uma solução bem mais curta e de fácil entendimento para quem lê o programa.

Terceira Forma - Substring de String

Cria-se um String contendo os 12 meses possíveis, enfileirando-se consecutivamente as 3 letras iniciais de cada mês e localiza-se o substring desejado por uma fórmula.

```
cMES = "JANFEVMARABRMAIJUN...DEZ";  
str := copy(cMES, (mes-1)*3+1, 3);
```

Nos dois primeiros exemplos, fica fácil verificar que funciona, embora seja preciso escrever bem mais. No terceiro exemplo, a escrita de programação é otimizada, sendo bem menor, porém o conteúdo de lógica é mais rebuscado.

Como saber se funciona? É aqui que entra o teste da solução...

Teste de Mesa

Este teste não precisa ser feito rodando-se o programa várias vezes. Basta a criação de uma pequena tabela (feita em Planilha Eletrônica, por exemplo):

Veja que o resultado da fórmula coincide com a primeira posição das 3 letras de cada mês!

Isto nos certifica de que o programa realizará exatamente o que dele se espera.

Conclusão

Conhecer bem uma boa linguagem de programação é importante mas não é tudo. Para termos programas com qualidade, que além de efetuar o trabalho desejado, facilitem a programação e a manutenção dos mesmos, é fundamental termos uma boa solução lógica do problema.

Meses Possíveis

Fórmula(mes- 1)*3+1

1	1
2	4
3	7
4	10
5	13
6	16
7	19
8	22
9	25
10	28
11	31
12	34

Paulo Sergio Borba, 33 anos, é Analista de Sistemas há 13 anos e Professor da Faculdade Radial São Paulo, onde leciona Linguagem e Técnicas de Programação no curso de Processamento de Dados.