

## Visualg: Uma Ferramenta de Auxílio à Programação - Parte II

Mário Leite

...

Como foi explicado na **Parte I**, o Visualg é uma ferramenta de apoio à Programação, oferecendo um ambiente integrado para desenvolver, editar, testar e depurar e executar programas em pseudocódigo. E a criação de um programa deve passar, necessariamente, por três etapas:

**Algoritmo:** criação da solução do programa através da descrição informal dos passos a seguir, de maneira bem livre, e sem se preocupar com qualquer tipo de formalismo computacional.

**Pseudocódigo:** tradução do Algoritmo, numa linguagem mais técnica, já considerando variáveis e estruturas de controle do fluxo, escritas em *portugol*.

**Código:** nesta terceira etapa é que o programador deve codificar (que muitos tratam como programar) o programa na linguagem que ele domina, dentro da sintaxe exigida pela linguagem.

A **figura 1** mostra um exemplo para calcular e exibir o resultado da divisão de um primeiro número lido pelo segundo lido depois. Observe que o programa “codificado” em Visualg é muito semelhante ao seu pseudocódigo. Esta semelhança não é mera coincidência, pois, a codificação dentro do Visualg segue, exatamente, o padrão *portugol*, que é uma mistura de Português com alguma sintaxe da ferramenta. Em pseudocódigo a grafia em Português é livre, podendo usar o til (~) em **Senão**, **Então**, e o cedilha em **Faça**. Já, no pseudocódigo do Visualg a grafia integral em Português não pode ser usada (a não ser em letras maiúsculas, e na versão mais atual). Entretanto, se o programador quiser, poderá criar seu pseudocódigo tal como o usado no Visualg. Aqui pode surgir uma pergunta: ‘ENTÃO, O PROGRAMADOR PODE SALTAR A ETAPA DE CRIAR O PSEUDOCÓDIGO E IR DIRETAMENTE PARA O AMBIENTE DO VISUALG?!’. Sim; claro que pode! Por isto, é que o Visualg é uma ferramenta de apoio ao aprendizado em programação, num ambiente de desenvolvimento de programas em pseudocódigo. MAS, a etapa de Algoritmo nunca pode ser desprezada, pois é a mais importante: é onde o programador cria a solução do problema. Assim, considerando o emprego do Visualg para criar o pseudocódigo e testar a solução dada no Algoritmo, a terceira etapa é a codificação do pseudocódigo numa linguagem real (VB, C, C#, Python, ...) para criar o chamado “código-fonte” do programa, o qual será executado pelo sistema operacional adequando.

---

### Algoritmo:

#### Início

- 1) **Leia** o primeiro número.
- 2) **Leia** o segundo número.
- 3) **Se** o segundo for diferente de zero **Vá** para o **Passo 4**, **Senão** volte ao **Passo 2**
- 4) **Divida** o primeiro número pelo segundo.
- 5) **Mostre** o resultado da divisão.

#### Fim

---

### Pseudocódigo

#### Programa "DivideNumero"

**Declare** Num, Den, Quoc: **real**

#### Início

**Leia**(Num)

**Leia**(Den)

**Enquanto** (Den=0) **Faça**

**Leia**(Den)

**FimEnquanto**

**Quoc** <- Num/Den

**EscrevaLn**(Quoc)

#### FimPrograma

---

### Em Visualg

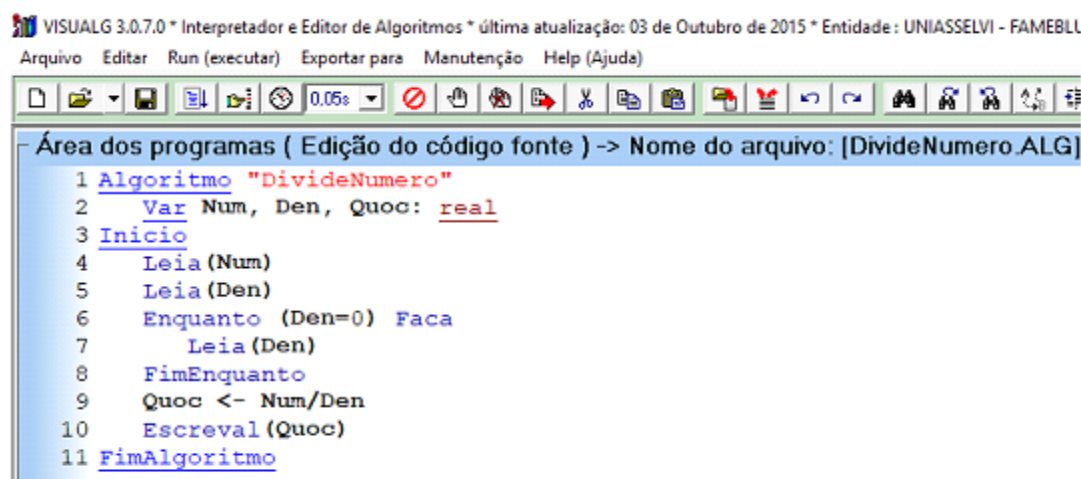


Figura 1 - As três etapas básicas de desenvolvimento de um programa