



---

# Software Visualg 2.0

---

**Bruno Tonet**

## Sumário

<b>1 - VISUALG - EDITOR E INTERPRETADOR DE ALGORITMOS .....</b>	<b>3</b>
1.1. OBJETIVOS.....	3
<b>2 - INSTALAÇÃO E REQUERIMENTOS DE HARDWARE.....</b>	<b>4</b>
<b>3 - A TELA PRINCIPAL DO VISUALG .....</b>	<b>4</b>
<b>4 - A BARRA DE TAREFAS .....</b>	<b>5</b>
<b>5 - A BARRA DE EXECUÇÃO DE ALGORITMO.....</b>	<b>6</b>
<b>6 - VISUALIZADOR DE VARIÁVEIS.....</b>	<b>8</b>
<b>7 - SIMULADOR DE SAÍDA.....</b>	<b>8</b>
<b>8 - A BARRA DE STATUS .....</b>	<b>9</b>
<b>9 - LISTA DE FUNÇÕES.....</b>	<b>9</b>
<b>10 - O MENU DO VISUALG .....</b>	<b>9</b>
10.1 ARQUIVO .....	9
10.2 EDITAR .....	10
10.3 EXIBIR .....	11
10.4 ALGORITMO.....	11
10.5 CÓDIGO .....	12
10.6 FERRAMENTAS.....	12
10.6.1 COMO UTILIZAR COMANDO LISTA .....	13
10.7 AJUDA .....	14

---

## 1 - Visualg - Editor e Interpretador de Algoritmos

---

### Sobre instituidor do Visualg

Cláudio Morgado de Souza é programador/analista e professor universitário no Rio de Janeiro. Atua na Área de desenvolvimento de software desde 1987, e no ensino de linguagens de programação e bancos de dados desde 1990. Atualmente é proprietário da Apoio Informática Ltda., em Valença, RJ, e leciona na Fundação Educacional D. Andre Arcoverde (FAA) em Valença, RJ e na Universidade Severino Sombra (USS) em Vassouras, RJ.

Sua área principal de interesse acadêmico é o ensino de linguagens de programação e algoritmos, e pode ser contatado no MSN Messenger ou por e-mail sob o endereço: cmorgado@apoioinformatica.inf.br.

---

### 1.1. Objetivos

---

A idéia para este programa nasceu da necessidade de uma ferramenta para os alunos iniciantes em programação exercitarem seus conhecimentos num ambiente próximo da realidade de uma linguagem de programação. Em minha experiência como professor desta disciplina tenho notado que a abstração de "rodar o chinês", executar um programa no papel apenas, é um grande obstáculo (quase intransponível para alguns) ao aprendizado das técnicas de construção de algoritmos. Por outro lado, submeter um iniciante aos rigores de uma linguagem como o Pascal ou ao esoterismo do C me parecia exagerado. O ideal seria uma linguagem parecida com o "Portugol", de grande popularidade nos meios acadêmicos e presente nos livros mais utilizados; com ela os princípios básicos da programação estruturada poderiam ser ensinados sem que a curva de aprendizagem fosse íngreme. O programa também deveria ser capaz de simular o que acontece na tela do computador com os famosos "leia" e "escreva", bem como possibilitar o exame dos valores de variáveis e suportar um modo simples de debugagem e acompanhamento passo a passo da execução do "programa" (pelo seu grande valor didático). Aliado a isto deveria estar um editor com recursos razoáveis, tais como salvar e abrir algoritmos, utilizando todos os recursos que o ambiente Windows provê.

O Visualg é para mim a concretização desta idéia, e espero que o colocando em domínio público possa ajudar professores e alunos de informática e ser ajudados por eles, com sugestões e críticas visando sempre a sua melhoria. A idéia básica é manter o Visualg simples; ele é como as rodinhas de apoio que uma criança usa ao aprender a andar de bicicleta: assim que elas não forem mais necessárias devem ser retiradas. Inicialmente, não achava necessário que houvesse recursos como subprogramas, recursividade, etc., pois quando o aluno estivesse a ponto de aprender estas técnicas já estaria trabalhando com uma linguagem real como Pascal, C ou Java. A aceitação do Visualg nos meios acadêmicos e a colaboração de vários professores e alunos tornaram esta implementação desejável e até necessária; a versão 2.0 já tem estes recursos. O Visualg ainda pode ser melhorado, é claro, e para isto conto com a colaboração de todos que vierem a utilizá-lo.

**Cláudio Morgado de Souza**

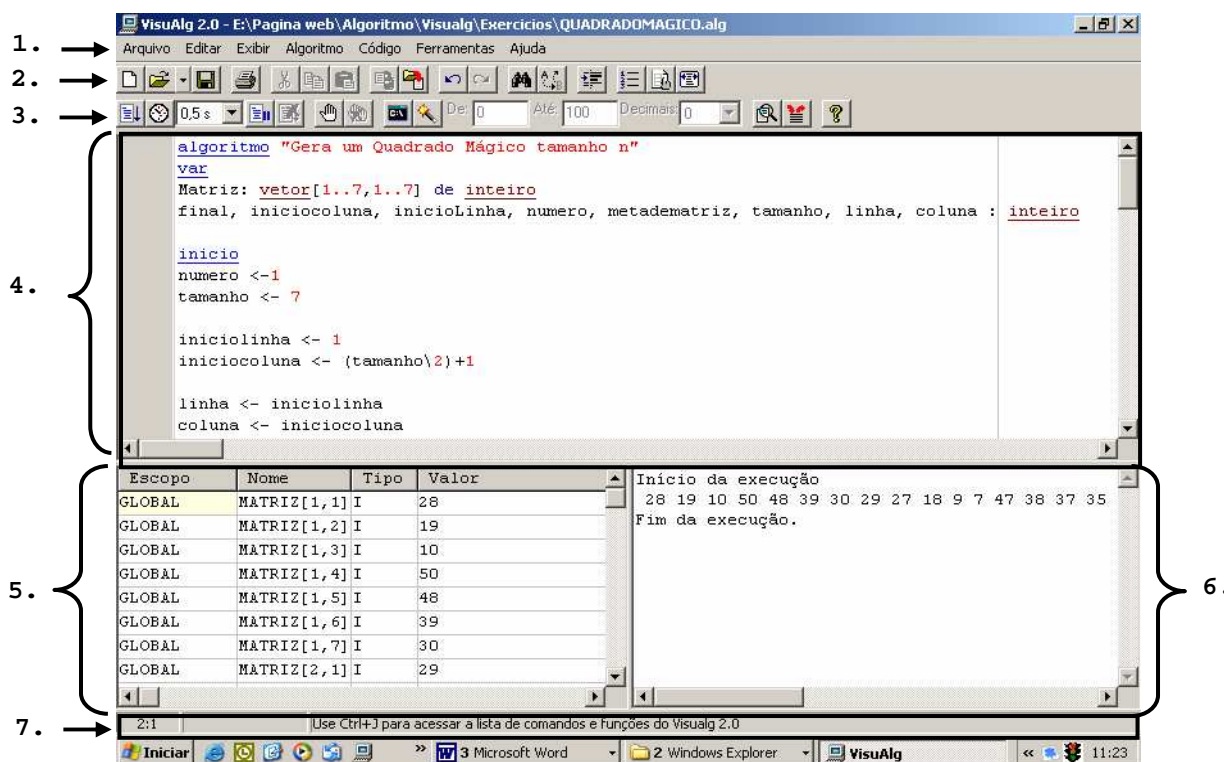
## 2 - Instalação e Requerimentos de Hardware

O Visualg é um programa simples, que não depende de DLLs, OCXs ou outros componentes. Sua instalação não copia arquivos para nenhuma outra pasta a não ser aquela em que for instalado, e toma em torno de 1 MB de espaço em disco. Pode ser executado sob o Windows 95 ou posterior, e tem melhor aparência com resolução de vídeo de 800x600 ou maior.

## 3 - A Tela Principal do Visualg

A tela do Visualg compõe-se:

1. Menu
2. Barra de tarefas
3. Barra de Execução de Algoritmo
4. Editor de textos (que toma toda a sua metade superior)
5. Visualizador de variáveis (no lado esquerdo da metade inferior)
6. Simulador de saída (no lado direito da metade inferior) e
7. Barra de status.





Quando o programa é carregado, já coloca no editor texto um "esqueleto" de algoritmo, cuja intenção é, além de poupar trabalho ao usuário, mostrar o formato básico de algoritmo que deve ser utilizado, bem como a forma dos comentários. A seguir explicamos cada componente da interface do Visualg.


## 4 - A Barra de Tarefas


Contém os comandos mais utilizados no Visualg (estes comandos também podem ser acessados pelo menu ou por atalhos no teclado).





 **Novo (Ctrl-N)** - Cria um novo "esqueleto" de algoritmo, substituindo o texto existente no editor. Se este tiver sido modificado, o Visualg pedirá sua confirmação para salvá-lo antes que seja sobreposto.

 **Abrir (Ctrl-A)** - Abre um algoritmo anteriormente gravado, substituindo o texto existente no editor. Se este tiver sido modificado, o Visualg pedirá sua confirmação para salvá-lo antes que seja sobreposto.


 **Salvar (Ctrl-S)** - Salva imediatamente o texto do editor. Caso seja a primeira vez que um novo texto é gravado, o Visualg pedirá o seu nome e localização.


 **Imprimir** - Imprime imediatamente o texto existente no editor. Para configurar a impressão, use o comando Imprimir... do menu Arquivo (acessível também pelo atalho *Ctrl-P*).


 **Recortar** - Move um texto selecionado para a memória.


 **Copiar** - Copia um texto selecionado para a memória.

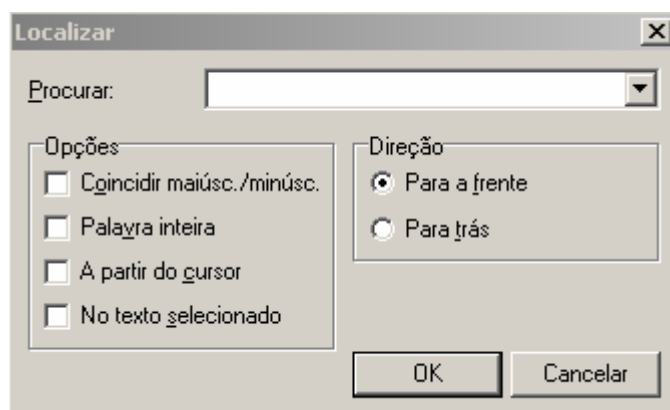
 **Colar** - Retira conteúdo da memória e coloca no local do cursor.


 **Gravar Bloco** - Abre uma janela para salvar o bloco selecionado. Objetivo é criar uma espécie de biblioteca de funções.

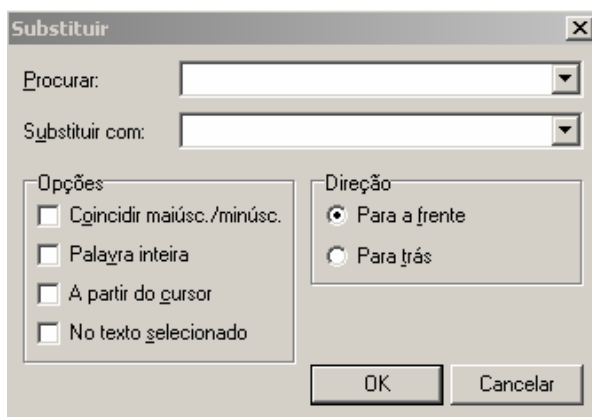
 **Inserir um bloco** - Insere um bloco arquivado no local do cursor.


 **Desfazer e refazer** - Desfaz e refaz ação criada no editor de texto.


 **Localizar** - Abre uma janela para você digitar a palavra que deseja localizar no Editor de textos.




 **Substituir** – Abre uma janela para você poder digitar a palavra que deseja localizar e substituir no Editor de textos.



 **Corrigir Indentação (Ctrl-G)** - Corrige automaticamente a indentação do "código-fonte", colocando os comandos dentro de uma estrutura de 3 colunas à direita da coluna inicial da estrutura conforme a configuração padrão.

 **Numerar Linhas** - Liga/desliga a exibição dos números das linhas na área à esquerda do editor. A linha e coluna do editor em que o cursor está em um determinado momento também são mostradas na primeira parte da barra de status, situada na parte inferior da tela. Os números de linhas, caso ligados, são desligados durante a execução do algoritmo por motivos técnicos, mas são ligados de volta ao fim do "programa".

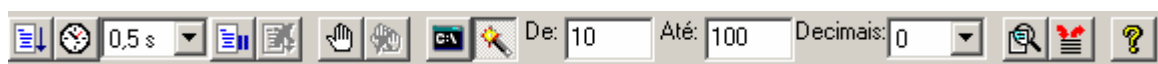
 **Mostra Variáveis Modificadas** - Liga/desliga a exibição da variável que está sendo modificada. Como o número de variáveis pode ser grande, muitas podem estar fora da janela de visualização; quando esta característica está ligada, o programa rola a grade de variáveis de modo que aquela que está sendo modificada no momento fique visível. Útil quando se está executando o algoritmo passo a passo. Por questões de performance, o valor padrão desta característica é *desligada* quando o algoritmo está sendo executado automaticamente, mas se você clicar este botão pode executar o algoritmo automaticamente com a exibição ligada. O valor volta automaticamente para *desligada* ao fim da execução.


 **Restaura a tela inicial do Visualg** – Retorna a tela original do programa.


---


## 5 - A Barra de Execução de Algoritmo

---



 **Executar (F9)** - Inicia (ou continua) a execução automática do algoritmo.

 **Tempo por linha (shift + F9)** – Executa o algoritmo linha por linha automaticamente determinado por um tempo escolhido pelo usuário.

 **Passo (F8)** - Inicia (ou continua) a execução do algoritmo linha por linha, dando ao usuário oportunidade de acompanhar o fluxo do programa, examinar variáveis, etc.



**Parar (Ctrl-F2)** - Termina imediatamente a execução do algoritmo. Este botão fica desabilitado quando o algoritmo não está sendo executado.



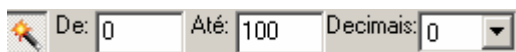
**Marca e Desmarca um Breakpoints (F5)** - Cria pontos de parada. Selecione a linha que deseja criar um ponto de parada na hora de execução do algoritmo e pressione o Breakpoints surgirá uma linha marrom e um marcador do lado esquerdo para indicar o ponto de parada. Para desmarcar o ponto de parada basta selecionar a linha que possui o breakpoint e clicar no mesmo. O breakpoints não funciona no modo passo a passo e para continuar o algoritmo depois de uma parada pressione novamente o **F9** ou o botão **Executar**.



**Desmarca todos os BreakPoints (CTRL+F5)**



**Executa em Modo Dos** - Quando ativado durante a execução do algoritmo ele executa o algoritmo em uma janela em modo dos.



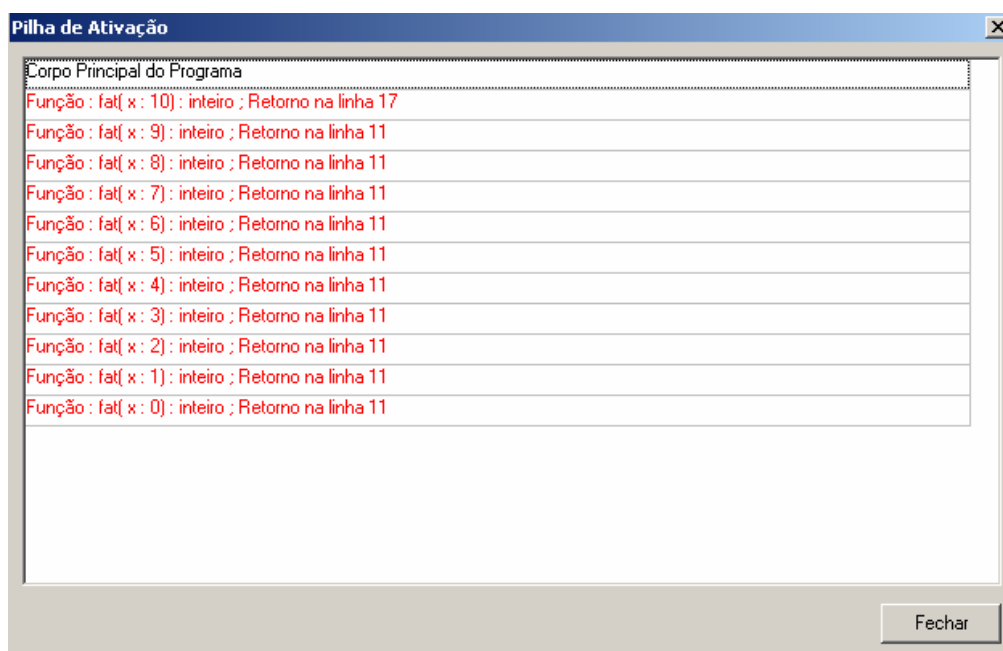
**Gerador valores** - Substitui digitação do usuário por um sistema de geração aleatória de valores numéricos e caracter (este comando não afeta a leitura de variáveis do tipo lógico - com certeza uma coisa pouco usual...). Gera números e caracteres aleatoriamente, muito útil para não perder tempo pensando o que digitar, você escolhe o início e o fim dos valores e se for valores com casas decimais é só escolher quantas casas decimal você quer.



**Perfil (F7)** - Mostra, após a execução de um algoritmo, quantas vezes cada linha foi executada. Útil para a análise de eficiência de um algoritmo, como por exemplo, nos métodos de classificação.



**Pilha (CTRL+F3)** - Mostra a pilha de ativação do programa (call stack), com o nome dos **procedimentos** e **funções** chamados, nome, tipo e valor dos parâmetros.



**Ajuda on-line (F1)** - Em construção.

## 6 - Visualizador de Variáveis

---

Contém uma grade onde são mostrados: **Escopo** da variável (Global quando nome da variável for global ou o nome da função ou Procedimento quando for local). O **Nome** da variável (com índice ou índices, caso seja um elemento de um vetor), seu **Tipo** ("R" para Real, "I" para Inteiro, "C" para literal e "L" para lógico), e o seu **Valor** corrente. A versão atual do Visualg permite até **500** variáveis (cada elemento de um vetor conta como uma variável).

Também, de acordo com o tipo de parâmetro a cor no grid muda, e para os parâmetros passados por referência. Há uma seta que mostra o nome da variável que eles representam fora do subprograma. Isto tudo, naturalmente, só pode ser visto se executarmos o algoritmo passo a passo...

Escopo	Nome	Tipo	Valor
GLOBAL	PAIS	C	Alemanha
GLOBAL	VALORTESTE	L	FALSO
TESTAPAIS	(RETORNO)	L	VERDADEIRO
TESTAPAIS	TESTENOME	C	Alemanha

---

## 7 - Simulador de saída

---

Mostra o resultado do algoritmo, invés de executar no modo dos, você pode verificar a saída do algoritmo aqui no simulador de saída.

```
Início da execução
70
75
35
39
24
17
68
21
48
3
Entre com o valor de busca (ESC termina) 39
Achei 39 na posição 4
Fim da execução.
```



## 8 - A Barra de Status

---

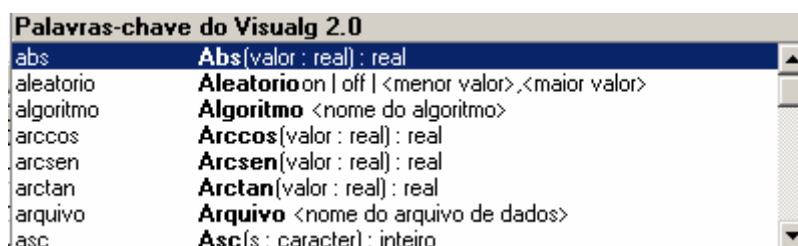


Situada na parte inferior da tela, contém três painéis. O primeiro mostra a linha e coluna onde o cursor está; o segundo mostra a palavra **Modificado** caso o algoritmo tenha sido alterado desde que foi carregado ou salvo pela última vez. O terceiro mostra o comando que mostra as funções predefinidas do Visualg.

## 9 - Lista de Funções

---

Lista de funções é uma maneira rápida de inserir funções predefinidas em seu algoritmo, a sua utilização é muito simples basta pressionar **CTRL+J** para mostrar a lista e selecionar a função desejada e dar um Enter, depois é só passar os parâmetros desejados.



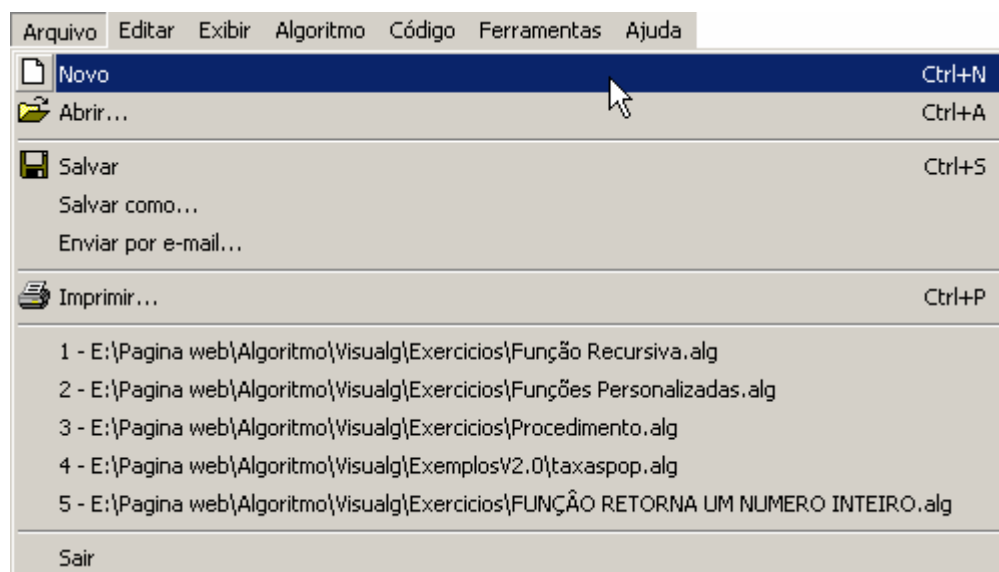
## 10 - O Menu do Visualg

---

### 10.1 Arquivo

---

Possui os comandos para se abrir, salvar e imprimir algoritmos:



**Novo** - Cria um novo "esqueleto" de algoritmo, substituindo o texto existente no editor. Se este tiver sido modificado, o VisuAlg pedirá sua confirmação para salvá-lo antes que seja sobreposto.

**Abrir** - Abre um algoritmo anteriormente gravado, substituindo o texto existente no editor. Se este tiver sido modificado, o VisuAlg pedirá sua confirmação para salvá-lo antes que seja sobreposto.

**Salvar** - Salva imediatamente o texto do editor. Caso seja a primeira vez que um novo texto é gravado, o VisuAlg pedirá o seu nome e localização.

**Salvar como ...** - Permite salvar o texto do editor, exibindo antes a janela para se escolher o nome e localização.

**Enviar por e-mail...** Permite mandar o algoritmo por e-mail.

**Imprimir...** - Permite a impressão do algoritmo corrente, mostrando antes a janela de configuração de impressão (o botão Imprimir da barra de tarefas imprime o algoritmo imediatamente na impressora padrão).

**Sair** - Abandona o Visualg.

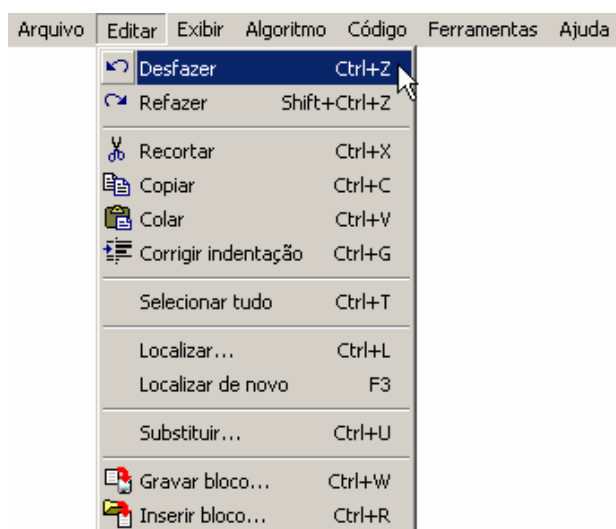
Além destes comandos, há ainda a lista dos 5 últimos algoritmos utilizados, que podem ser abertos diretamente ao se escolher o seu nome.

---

## 10.2 Editar

---

Possui os comandos de manipulação de textos:



**Desfazer** – Desfaz a última ação feita no editor de texto.

**Refazer** - Refaz a última ação desfeita no editor de texto.

**Recortar** – Move um texto selecionado para a memória.

**Copiar** – Copia um texto selecionado para a memória.

**Colar** – Retira conteúdo da memória e coloca no local do cursor.

**Corrigir Indentação (Ctrl-G)** - Corrige automaticamente a indentação do "código-fonte", colocando os comandos dentro de uma estrutura 3 colunas à direita da coluna inicial da estrutura.

**Seleciona tudo** – Marca todo o texto no editor de texto.

**Localizar** – Abre uma janela para você digitar a palavra que deseja localizar no Editor de textos.

**Localizar de novo (F3)** – Procura pela a ultima palavra localizada.

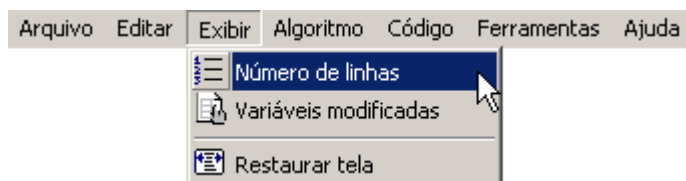
**Substituir** – Abre uma janela para você poder digitar a palavra que deseja localizar e substituir no Editor de textos.

**Gravar um bloco (Ctrl+W)** – Salva um bloco selecionado. Objetivo é criar uma espécie de biblioteca de funções.

**Inserir um bloco (Ctrl+R)** – Insere um bloco salvo no local do cursor.

### 10.3 Exibir

Possui os comandos para ligar/desligar as seguintes características:



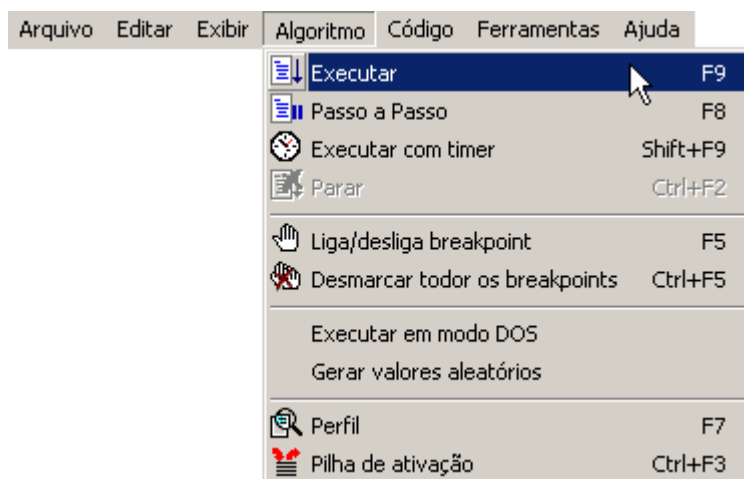
**Número de linhas** - Liga/desliga a exibição dos números das linhas na área à esquerda do editor. A linha e coluna do editor em que o cursor está em um determinado momento também são mostradas na primeira parte da barra de status, situada na parte inferior da tela. Os números de linhas, caso ligados, são desligados durante a execução do algoritmo por motivos técnicos, mas são ligados de volta ao fim do "programa".

**Variáveis modificadas** - Liga/desliga a exibição da variável que está sendo modificada. Como o número de variáveis pode ser grande, muitas podem estar fora da janela de visualização; quando esta característica está ligada, o programa rola a grade de variáveis de modo que aquela que está sendo modificada no momento fique visível. Útil quando se está executando o algoritmo passo a passo. Por questões de performance, o valor padrão deste item é *desmarcado* quando o algoritmo está sendo executado automaticamente, mas se você clicá-lo pode executar o algoritmo automaticamente com a exibição ligada. O item volta para *desmarcado* ao fim da execução (ele está relacionado ao botão correspondente da barra de tarefas).

**Restaura tela** – Retorna a tela original do programa visualg.

### 10.4 Algoritmo

Contém os comandos relativos à execução do algoritmo:



**Executar** - Inicia (ou continua) a execução automática do algoritmo.

**Passo a passo** - Inicia (ou continua) a execução do algoritmo linha por linha, dando ao usuário oportunidade de acompanhar o fluxo do programa, examinar variáveis, etc.

**Executar com tempo** – Executa o algoritmo linha por linha automaticamente determinado por um tempo escolhido pelo o usuário.

**Parar** - Termina imediatamente a execução do algoritmo. Este item fica desabilitado quando o algoritmo não está sendo executado.

**Marca e Desmarca um Breakpoints (F5)** – Cria pontos de parada. Selecione a linha que deseja criar um ponto de parada na hora de execução do algoritmo e pressione o

Breakpoints surgirá uma linha marrom e um marcador do lado esquerdo para indicar o ponto de parada. Para desmarcar o ponto de parada basta selecionar a linha que possui o breakpoint e clicar no mesmo. O breakpoints não funciona no modo passo a passo e para continuar o algoritmo depois de uma parada pressione novamente o **F9** ou o botão **Executar**.

**Desmarca todos os BreakPoints (CTRL+F5)**

**Executa em Modo Dos** – Quando ativado durante a execução do algoritmo ele executa o algoritmo em uma janela em modo dos.

**Gerar valores aleatórios** – Substitui digitação do usuário por um sistema de geração aleatória de valores numéricos e caracter (este comando não afeta a leitura de variáveis do tipo lógico - com certeza uma coisa pouco usual...), Gera números e caracteres aleatoriamente, muito útil para não perder tempo pensando o que digitar, você escolhe o início e o fim dos valores e se for valores com casas decimais é só escolher quantas casas decimal você quer.

**Perfil** - Mostra, após a execução de um algoritmo, quantas vezes cada linha foi executada. Útil para a análise de eficiência de um algoritmo, como por exemplo, nos métodos de classificação.

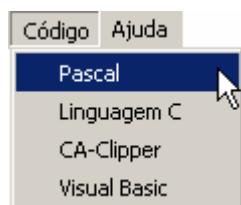
**Pilha de ativação (CTRL+F3)** - Mostra a pilha de ativação do programa (call stack), com o nome dos **procedimentos** e **funções** chamados, nome, tipo e valor dos parâmetros.

---

## 10.5 Código

---

Contém os comandos relativos ao "código-fonte":



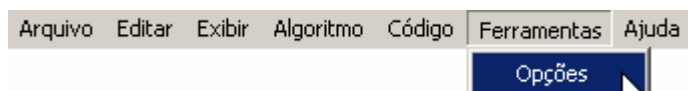
**Pascal, Linguagem C, CA-Clipper, Visual Basic** - Em futuras versões do Visualg, este menu conterà também os comandos para geração de código-fonte nas linguagens especificadas, a partir do algoritmo corrente. Atualmente ele gera apenas o pascal.

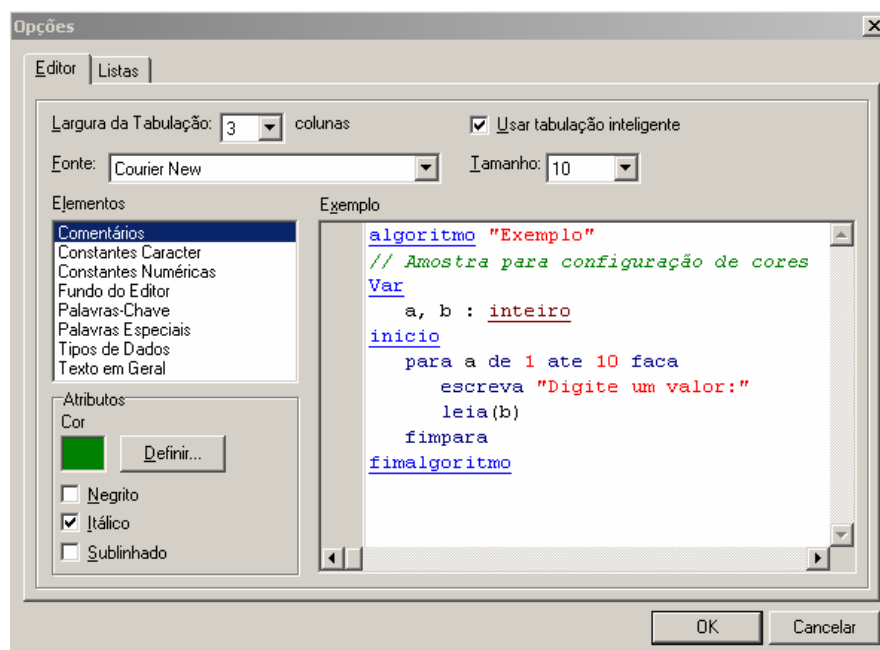
---

## 10.6 Ferramentas

---

Mostra duas abas, aba Editor onde você pode personalizar a fonte, cor, tabulação do Visualg e a aba Lista que você pode criar uma lista de dados para serem inseridas nas variáveis.



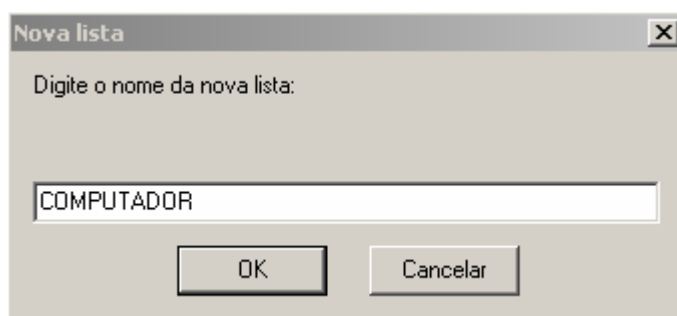


---

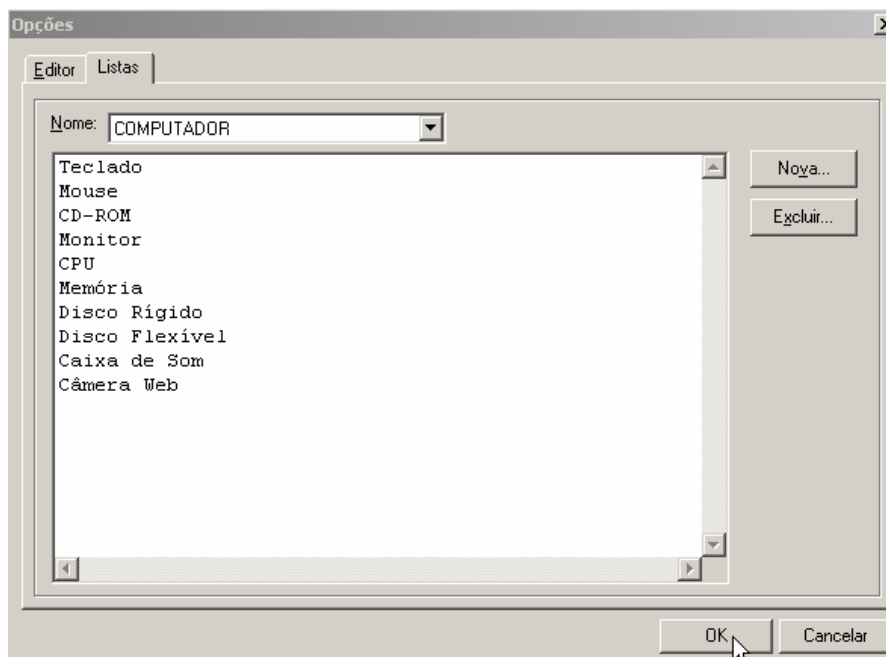
## 1.1. Como utilizar comando lista


---

Clicando no botão novo irá aparecer uma janela pedindo o nome da lista. Digite o nome desejado e de um OK.



Em seguida digite a lista de dados que deseja que apareça nas variáveis.



No algoritmo antes do comando leia coloque o seguinte comando: **LISTA "Computador"** e deixe a opção  **Gerar valores aleatórios**. Veja um exemplo:

```
1.  Algoritmo "Comando Lista"
2.  var
3.  Computador,Pessoas :caracter
4.  inicio
5.  LISTA "Computador"
6.  Leia (Computador)
7.  ESCREVAL (Computador)
8.  LISTA "Pessoas"
9.  Leia (Pessoas)
10. ESCREVAL (Pessoas)
11. fimalgoritmo
```

## 10.7 Ajuda

Contém o acesso às páginas de ajuda do Visualg e à janela Sobre.

