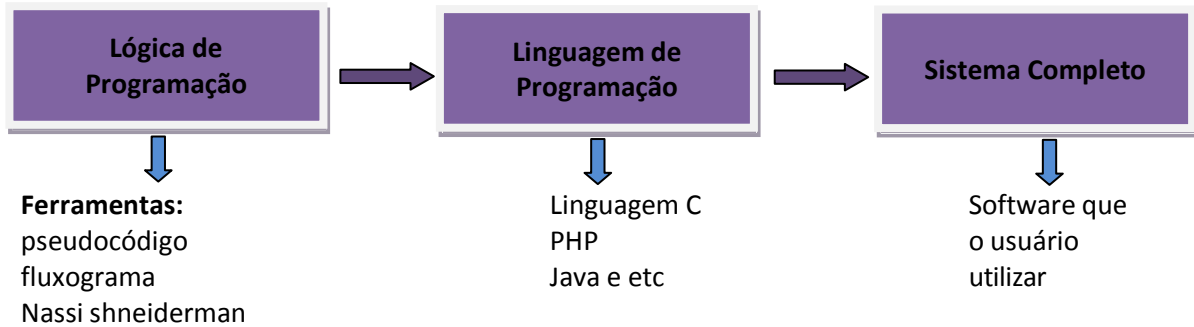


## Algoritmos Computacionais

São passos a serem seguidos por um dispositivo processador (tudo aquilo que pode realizar processamento, computador pessoal, tablet, celular...) e seus respectivos usuários que, quando executados na ordem correta, conseguem realizar determinada tarefa.

**Usuário:** pode ser você mesmo ou outras pessoas que utilizam o sistema.

### Primeiro passo para criar algoritmos computacionais:



### Exemplo de algoritmo de uma rotina:

1ª Versão

algoritmo **atravesarRua**

olhar para a esquerda  
olhar para direita  
**se** estiver vindo um carro  
    não atravessar

**senão**

    atravessar

**fimse**

fimalgoritmo

### Observação:

Os algoritmos são lidos de cima para baixo e da esquerda para direita.

2ª Versão

algoritmo **atravesarRua**

olhar para a direita  
olhar para esquerda  
**se** não estiver vindo um carro  
    atravessar

**senão**

    não atravessar

**fimse**

fimalgoritmo

Um algoritmo é um conjunto de instruções finitas colocadas em uma sequência lógica, com o objetivo de resolver um determinado problema. Podemos ver no exemplo a seguir um exemplo incorreto da aplicação lógica.

3ª Versão

algoritmo **atravesarRua**

    atravesse  
**se** estiver vindo um carro  
        olhar para direita

**senão**

olhar para esquerda  
fimse  
não atravesse

finalgoritmo

Para estruturar, representar a lógica de programação utilizamos o pseudocódigo (português estruturado) ou português. Pseudocódigo é a lógica da programação escrita na linguagem nativa, no nosso caso por ser em português ficou batizado como português.

Não confundir, português, pseudocódigo (português estruturado) não é uma linguagem de programação, é uma maneira de representar a sua lógica de programação antes de passar para uma linguagem de programação (linguagem c, java, php...). Tendo o português ou pseudocódigo (português estruturado) podemos converter essa lógica para qualquer programação.

A ferramenta que vamos utilizar para lógica de programação é o VISUALG, visualizador de algo.

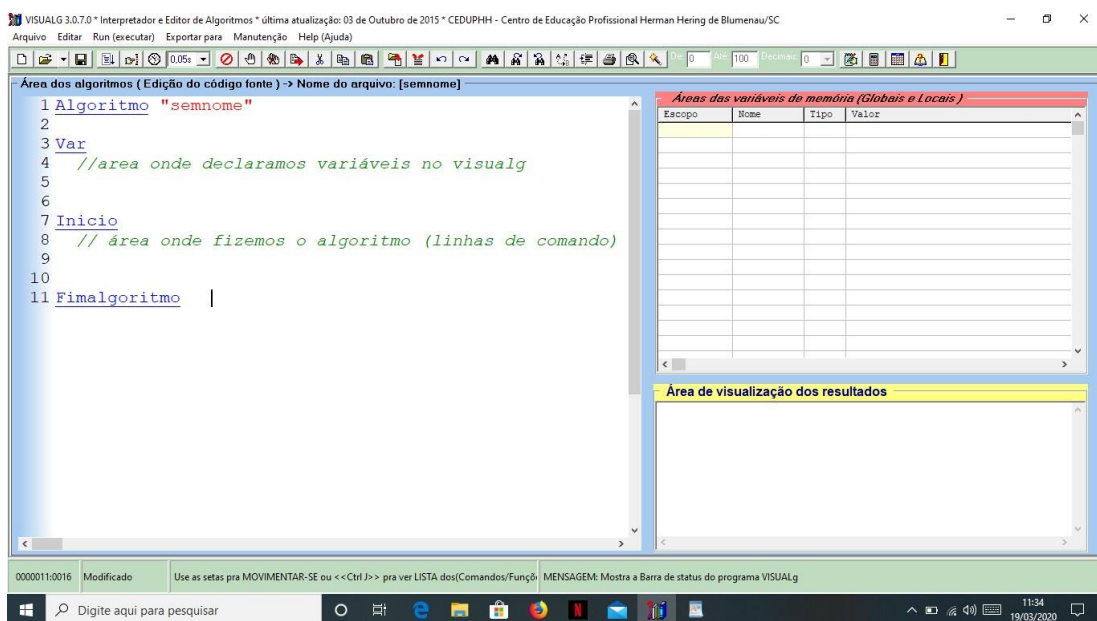
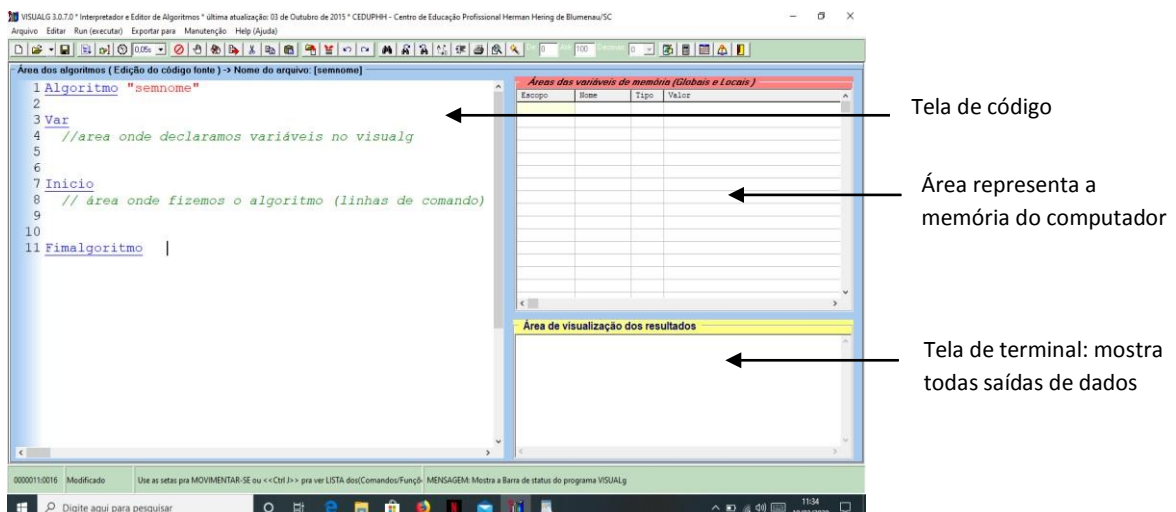


Figura 1 - ferramenta visualg

Áreas da tela do visualg:



## Comandos de Saída

escreva()

escreval()

Os dois comandos mostram algo na tela, a única diferença é que escreval() imprime na tela e após faz a quebra de linha, ou seja, dá um ENTER.

escreva(" ")

↑ tudo que está entre aspas significa uma mensagem, um caractere ou cadeia de caracteres.

A seguir um exemplo do uso do comando escreva().

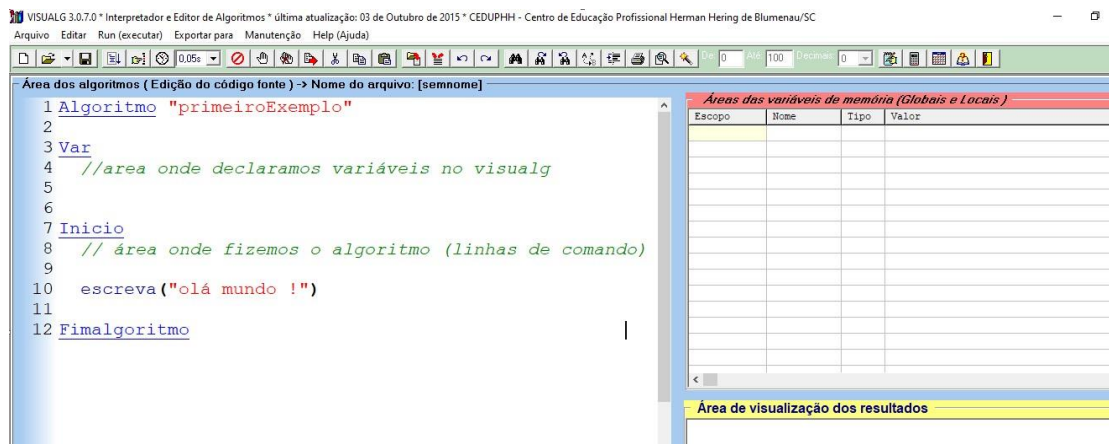




Figura 2

Conforme figura 2, mostra que devemos colocar o comando na tela de código, entre a palavra **inicio** e **fimalgoritmo** (onde declaramos o português estruturado), nessa área.

Para executar o código ir na barra de menu, Run (executar), ou tecla F9 ou clique em  ou clique em .

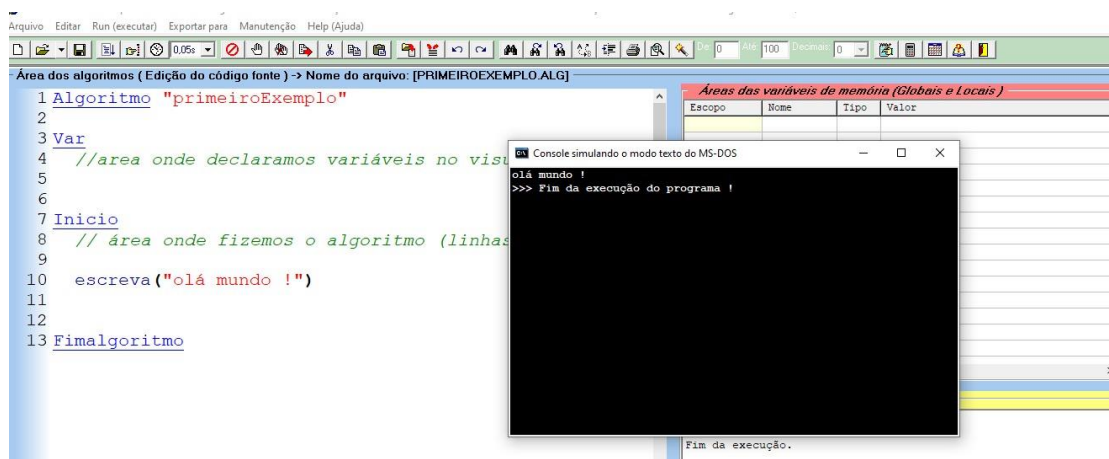


Figura 3

A figura 3 mostra que após executar o algoritmo apareceu a mensagem olá mundo. Essa é a saída do algoritmo primeiroExemplo.

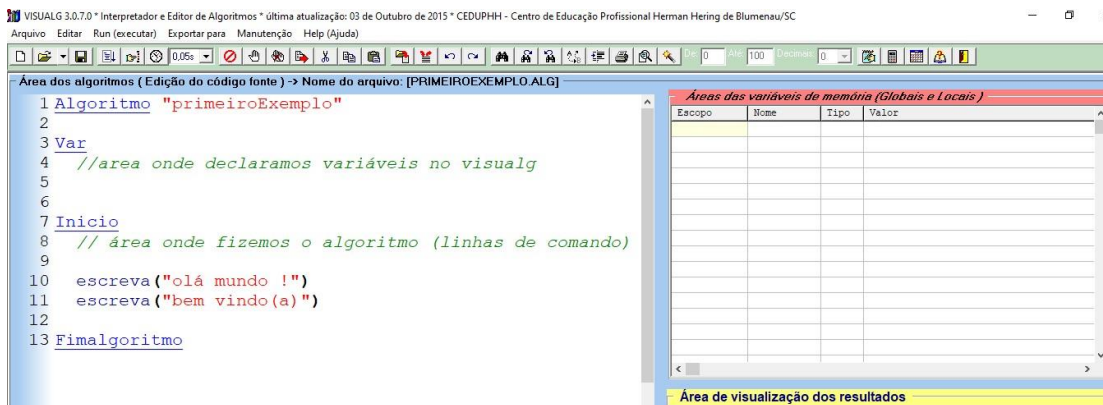


Figura 4

Na figura 4, onde foi inserido mais uma linha de comando para o computador mostrar, além de olá mundo a mensagem bem vindo(a).

Após executar:

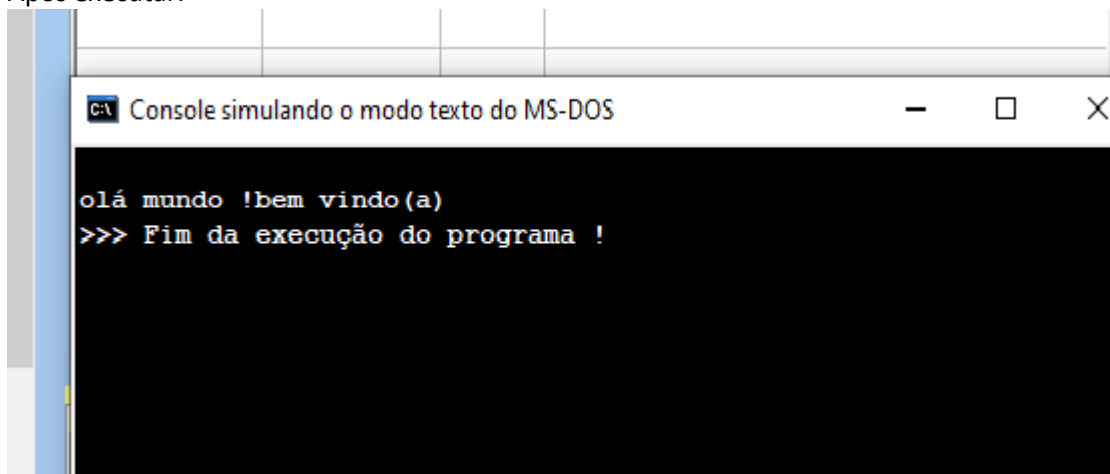


Figura 5

Na figura 5, a professora resolveu só mostrar a área de tela de terminal, onde mostra todas as saídas, no caso as mensagens dos dois comandos escreva(). Como podem observar as saídas dos 2 comandos escreva, ficaram na mesma linha. Isso porque o comando escreva não faz a quebra de linha conforme já foi explicado anteriormente.

Para fazer que cada mensagem fique em uma linha observem a figura 6 a seguir:

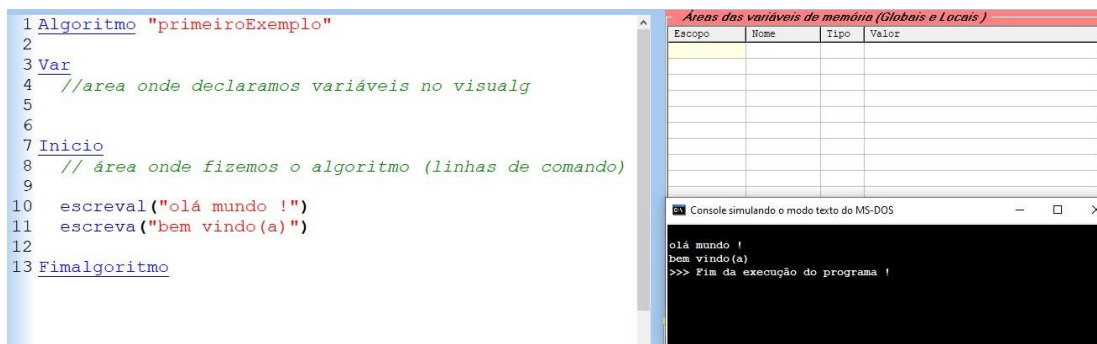
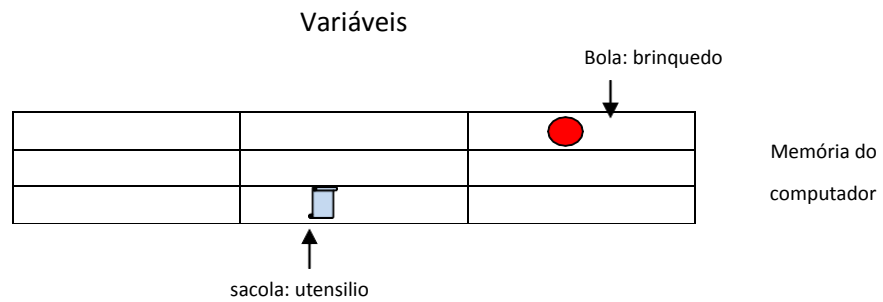


Figura 6

No primeiro comando foi colocado o comando de saída escreval(), que imprime e após faz a quebra de linha.



Variável

Armazenamento de dados na memória do computador.

var

identificador: tipo de dados

**Existe 6 regras para um identificador:**

1. Deve começar com uma letra
2. As próximas podem ser letras ou números
3. Não pode usar nenhum símbolo, exceto
4. Não podem conter espaços em branco
5. Não podem conter letras com acentos
6. Não podem ser uma palavra reservada

Exemplo de identificadores válidos:

nota1	media
salario	salarioBruto
saldoBancario	salario_liquido
media_salarial	salario_bruto

**Todos os dados armazenados em um identificador (variáveis) são classificados do tipo:**

**inteiro** - vai permitir que a sua variável guarde apenas inteiros. São aqueles não fracionados.

Ex.: ... -3, -2, -1, 0, 1, 2, 3, 4, 5...

**real** - vai permitir que a sua variável guardem apenas números fracionáveis.

Ex.: -3.0, -2.5, -1.0, 0.0, 1.0, 1.5, 2.0, 2.2....

**caractere** - é tudo aquilo que é colocado entre aspas.

Ex.: "Sandra", "olá Mundo!", "123", "algoritmo"

**lógico** - armazena apenas verdadeiro ou falso

Ex.: ligado = verdadeiro ou ligado = falso

**Comando de atribuição**

No visualg, os operadores de atribuição são:

:= dois pontos e sinal de igual

<- sinal de menor e igual

**Uso de variáveis:**

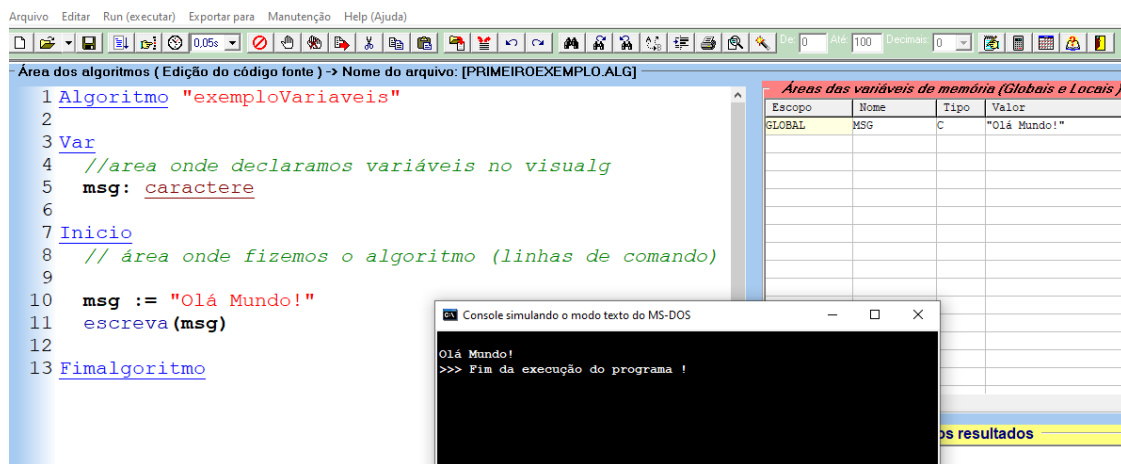


Figura 7

Na figura 7 esta declarada a variável msg do tipo caractere. No algoritmo existe 2 linhas de comandos.

Na primeira linha de código temos msg := "Olá Mundo!", **que se lê**, msg recebe Olá Mundo!. Ou seja na variável msg está armazenado a partir dessa linha informações do tipo cadeia de caracteres.

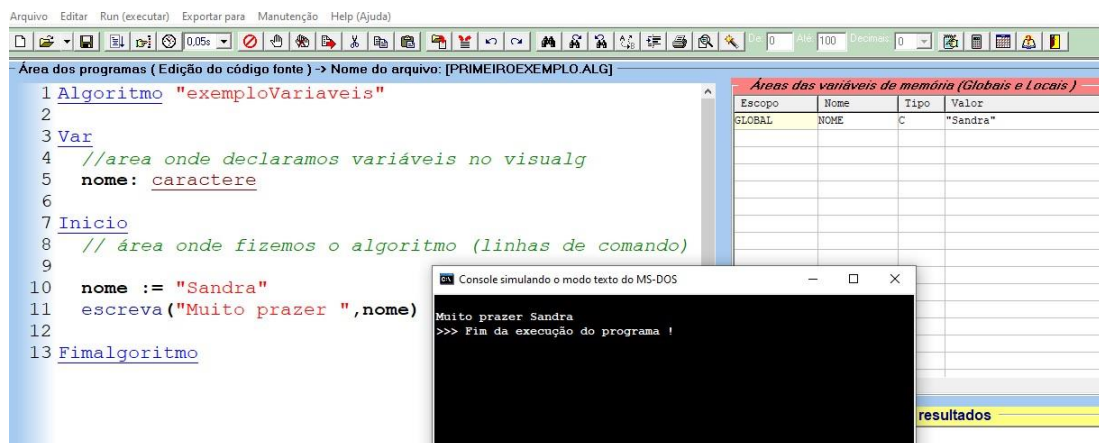


Figura 8

Na figura 8, esta declarada a variável nome do tipo caractere. No algoritmo existe 2 linhas de comandos.

Na primeira linha de código temos nome := "Sandra", **que se lê**, nome recebe Sandra. Ou seja, na variável nome está armazenado a partir dessa linha informações do tipo cadeia de caracteres.

Na segunda linha de código, o comando escreva("Muito prazer ", nome) está indicando que mostre duas informações: primeiro a mensagem Muito prazer e segundo o conteúdo da variável nome.

### Como interagir com o usuário

Comando de entrada de dados:

leia() ou seja, o comando permite entrada de dados na variável.

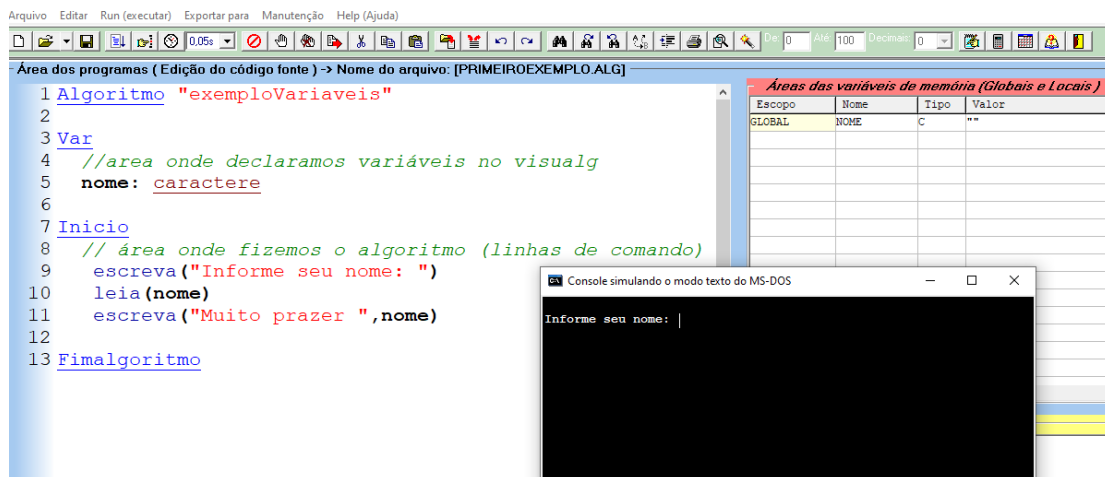


Figura 9

Na figura 9, esta declarada na variável **nome** do tipo caractere. No algoritmo existe 3 linhas de comandos.

Na primeira linha, o comando `escreva()`, mostra uma mensagem para o usuário na tela, conforme mostra a figura. Essa mensagem é importante para o usuário saber o que o programa esta solicitando no momento.

Na segunda linha, o comando `leia(nome)`, recebe a partir do momento que o usuário digitar a resposta para mensagem, e dar ENTER, o conteúdo, ou seja, a informação que ficará armazenada nela em tempo de execução do algoritmo.

Na terceira linha de código, o comando `escreva("Muito prazer ",nome)` está indicando que mostre duas informações: primeiro a mensagem Muito prazer e segundo o conteúdo da variável nome.

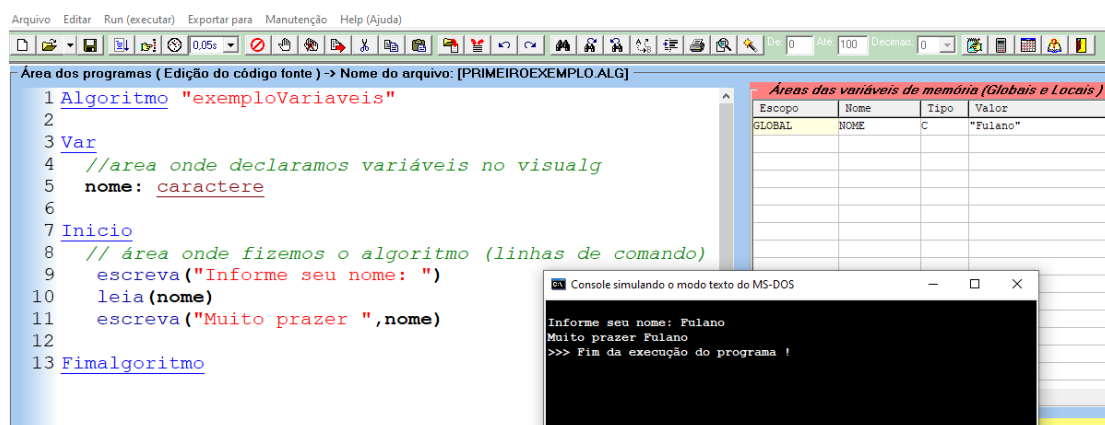


Figura 10

Na figura 10, podemos ver o resultado, ou seja, a execução completa do algoritmo acima.

## Operadores Aritméticos

Operadores Aritméticos	a := 5 b := 2	
+ adição	a + b	7
- subtração	a - b	3
* multiplicação	a * b	10
/ divisão	a / b	2.5

\ divisão inteira	a\b	2
^ exponenciação	a^b	25
% módulo	a % b	1

### Ordem de precedência

- ( ) parênteses
- ^ exponenciação
- \* ou / multiplicação ou divisão
- + ou - adição ou subtração

**Observação:** Com base no que vocês aprenderam segue exercícios de revisão do conteúdo visto até aqui.

- 1) Elabore um algoritmo que solicite para o usuário inserir dois valores inteiros e mostre na tela ao executar o algoritmo a soma desses dois valores.
- 2) Elabore um algoritmo que leia o nome do usuário e a qualidade dessa pessoa, e exiba na mensagem "< nome > é uma pessoa que tem <qualidade>".