

Outro ponto que precisa ser destacado é que, dentro das sub-rotinas, pode ocorrer declaração de variáveis, chamadas variáveis locais. Elas recebem este nome porque podem ser utilizadas apenas dentro da sub-rotina. Quando a execução desta chega ao fim, essas variáveis são destruídas e seus conteúdos são perdidos.

Variáveis declaradas fora de qualquer sub-rotina são chamadas de globais. Elas recebem este nome porque qualquer ponto do programa, incluindo as sub-rotinas, pode utilizá-las. São destruídas quando a execução do programa chega ao fim.

**OBSERVAÇÃO** Não se aconselha a utilização excessiva de variáveis globais, por tornar difícil a manutenção e a busca por erros nos programas.

## 8.2 SUB-ROTINAS EM PASCAL (*PROCEDURES E FUNCTIONS*)

A linguagem Pascal possibilita a modularização por meio de *procedures* (procedimentos) e *functions* (funções).

### 8.2.1 *PROCEDURES* SEM PASSAGEM DE PARÂMETROS

As *procedures* (procedimentos) são rotinas chamadas pelo programa principal para executar alguma operação específica, mas não retornam valor para quem as chamou. Possuem a seguinte sintaxe:

```
PROCEDURE nome da procedure;
declaração de variáveis locais;
BEGIN
    comandos;
END.
```

Quando o programa encontra uma linha contendo o nome da *procedure*, o fluxo da execução é desviado para as linhas contidas dentro dela. Esta execução só retornará ao fluxo normal quando a execução da *procedure* chegar ao fim.

A seguir é apresentado um exemplo de *procedure* sem parâmetros (a numeração das linhas não faz parte do programa).

```
1  PROGRAM EXEMPLO;
2  USES CRT;
3  VAR I, P, NUM, CONT : INTEGER;

4  PROCEDURE PAR;
5  {Conta quantos n°s pares existem entre 1 e o
   ➤ número digitado no programa principal}
6  BEGIN
7  CONT:=2;
8  P:=0;
9  WHILE CONT <= NUM DO
10     BEGIN
11     P:=P+1;
12     CONT:=CONT+2;
13     END;
14  END;

15  PROCEDURE IMPAR;
16  {Conta quantos n°s ímpares existem entre 1 e o
   ➤ número digitado no programa principal}
17  BEGIN
18  CONT:=1;
19  I:=0;
20  REPEAT
21     I:=I+1;
22     CONT:=CONT+2;
23  UNTIL CONT > NUM;
24  END;
```