**Proceduri**

(1) **Procedura** este un subprogram care efectuează prelucrarea datelor comunicate în momentul apelului. Limbajul conține preocedurile predefinite *read, readln, write, wirteln ș.a*. În completare, programatorul poate defini proceduri proprii, care se apelează în același mod ca procedurile-standart. Prin urmare, conceptul de procedură extinde noțiunea de ***insturcțiune*** PASCAL.

(2) Procedura poate să întoarcă mai multe rezultate, dar nu prin numele ei, ci prin variabile desemnate special (cu prefixul **var**) în lista de parametri formali.

(2) Parametri din listă introduși prin delcarații de forma **\*slide\*** se numesc **parametri-valoare**. Aceștia servesc pentru transmiterea de valori din programul principal in procedură.

(2) Parametri formali introduși în listă prin declarații de forma **\*slide\*** se numesc **parametri-variabilă** și servesc pentru întoarcerea rezultatelor din procedură în programul principal.

(3) **Activarea unei proceduri** se specifică printr-un apel de forma **\*slide\*** cu ***p*** numele procedurii; ***(a1, . . . ,an)*** lista (opțională) de parametri actuali. Spre deosebire de funcție, apelul de procedură este o instrucțiune; aceasta se inserează in program în locul în care sunt dorite efectele produse de execuția procedurii pentru eventualii **parametri** **actuali** ***a1, . . . ,an***.

(3) Să observăm fiecare element al unei proceduri într-un exemplu concret, un program ce calculează suma elementelor unui tablou.

(4) Putem observa că structura procedurii se aseamană cu ce a unui program simplu, singura diferență fiind antetul în care este declarată procedura.