

ACTOP

This subroutine is used in conjunction with PRAMCH to check the legality of an actual parameter and to compile the appropriate object program instruction.

A count of actual parameters in the current procedure call is kept in PRMCOU which is checked, in the routine '))' against the number of parameters in the namelist entry for the procedure name.

If E is set to expression level, this actual parameter could be an expression or a procedure call.

Example

- (i) PROCALL (a + Q (s) + b,
- (ii) PROCALL (a, Q (s),

The difference between these cases is shown by the top of the stack; in the latter case the top of the stack is '(' since each actual parameter unstacks back to the '(' of its procedure call; in the case of an expression, the top of the stack will be an arithmetic, relational or logical operator. If this actual parameter is an expression, TAKE and UNSTAK are used to complete the processing of the preceding expression, E is reset to 1, and PRAMCH is called to check that the corresponding formal parameter is a scalar called by value, and to compile a type conversion if necessary.

If this actual parameter is a procedure call, PRAMCH is used to check that the corresponding formal parameter is a scalar called by value.

If E is set to statement level, the last delimiter (LASTDL) is examined to determine between the three possible types for this actual parameter.

Example

- (i) PROCALL (a, B[1],LASTDEL =]
 - (ii) PROCALL (a, {string},LASTDEL = }
 - (iii) PROCALL (a,b,cLASTDEL = ,
- or (

In the first case, the top of the stack is used to determine whether this is an array or switch subscript. If the former, PRAMCH is used to check that the formal parameter is a scalar called by value, and to compile a type conversion if necessary. If the latter, the top of the stack is GTS or GTFS and PRAMCH is used to check that the formal parameter is a label called by value.

In the second case, PRAMCH is used to check that the corresponding formal parameter is a string called by name.

ACTOP (Contd)

In the third case, M is tested to determine whether the actual parameter is an identifier or a constant. If M = 1, this is an identifier and the Namelist is searched to find the declared entry. PRAMCH then checks the type of the actual parameter with that of the formal parameter and compiles the relevant object program operations.

If M = 2, this actual parameter is a constant and SECODL is used to find or enter the constant in the constant list, and PRAMCH is used to compile the correct "Take constant" operation.

Calls of PRAMCH:

0	Identifier
1	Constant
2	Expression
3	String
4	Switch call
5	Array subscript or procedure call.

ERRORS

FAIL 6 ; More than 14 parameters

FAIL 5 ; Illegal actual parameter

FAIL 49 ; Blank parameter