



Tech Info Library

Pascal II: Operand Formats (4 of 4)

Booleans:

The Boolean (binary) variable can have two values: TRUE and FALSE. Booleans are most commonly used in determining yes/no conditions, such as equality or set inclusion. Boolean variables are stored in one word, though only the LSB (least significant bit) is used. TRUE is represented by a 1; FALSE is represented by a 0.

MSB	15	8	7	0	LSB
																	Boolean

UCSD Pascal does not allow direct printing of Boolean variables. For example:

```

Program PrintBoolean;
Var A: boolean;
Begin
  A := FALSE;
  Writeln (A);    (* this is illegal *)
  If A = FALSE Then Writeln ('FALSE') Else Writeln ('TRUE');
  (* this is correct *)
End.

```

Booleans are most efficient in packed arrays, where each bit of the word is utilized. DrawBlock is probably the best-known example of this use. An excellent example of the use of boolean packed arrays is in the GrafDemo program on the Apple Pacal diskette APPLE3.

Boolean variables may be passed by value or by address.

Other Types:

In addition to all the above standard types, Pascal allows the programmer to define a wide variety of non-standard variable types. Probably the most popular example of this is the SET.

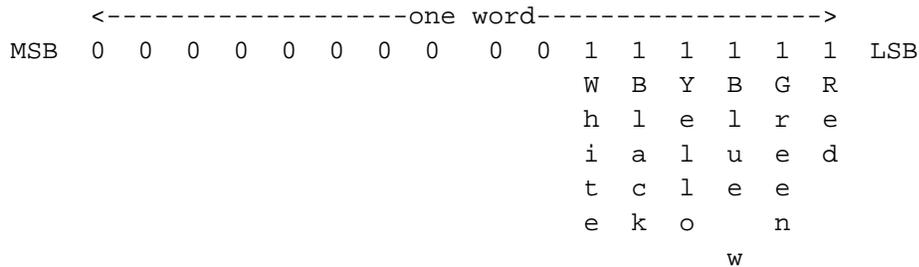
A set is an arbitrary collection of elements with each element assigned an ordinal position (that is, represented by a number). Each element of the set is represented by a name; you may choose any word for this name, except for (a) words reserved by Pascal, and (b) other variable definitions already in use. Each name is then associated with one bit in the data definition, beginning with bit 0. The set is stored in memory as a series of bits identified by the ordinal position of the element in the type definition. A set must end on a

word boundary: for example, 17 elements would take up 2 words, even though only one bit of the second word is actually used.

Example:

```
Type Colors = (Red,Green,Blue,Yellow,Black,White);
ColorSet = Set of Colors;
```

is a set of colors. Red occupies position 0, and white has position 5.



Sets may be passed either by address or by value, with certain restrictions. See p. 203 of the Pascal reference manual for details.

In general, complex record types consist of one or more standard types, each stored as described above. For the last word on Pascal data types, read Niklaus Wirth's Report in "User Manual and Report" by Jensen and Wirth.

References:

- Apple Pascal Reference Manual, by Apple Computer Inc. 1979.
 - Apple Pascal Language Reference Manual, by Apple Computer, 1980.
 - Apple Pascal Operating System Reference Manual, by Apple Computer, 1980.
 - Programming in Pascal, by Peter Grogono, Addison Wesley, 1978.
 - User Manual and Report, by Kathleen Jensen and Niklaus Wirth, Springer-Verlag, 1974.
- Apple Tech Notes

Keywords: <None>

=====

This information is from the Apple Technical Information Library.

19960215 11:05:19.00

Tech Info Library Article Number: 689