

PEOPLE WHO HAVE WRITTEN LOTS OF CODE ON OLD-STYLE TELETYPES OR CARD-PUNCHES OFTEN TEND TO WRITE VARIABLE NAMES, LABELS AND COMMENTS IN BLOCK CAPITALS. THAT'S FINE AS LONG AS ALL COMMENTS AND VARIABLE NAMES HAVE AN EQUAL EMPHASIS; IF THEY DO NOT IT IS NECESSARY TO RESORT TO A FEW RATHER

```
#####
#
#          UNUSUAL          #
#
#####
SYNTACTIC TRICKS TO DRAW SPECIAL ATTENTION TO WORDS THAT MUST STAND OUT.
THERE IS ALSO A PROBLEM WITH TELLING WHERE ONE SENTENCE ENDS AND ANOTHER
BEGINS, AND IT IS SORT OF DIFFICULT TO USE VERYLONGVARIABLENAMES WITHOUT
UNDUE CONFUSION.
```

There was once a time when names could have no more than six letters. This limit created serious problems for creative expression of thoughts. There were essentially two solutions: abbreviation and truncation. On the other hand, fewer keywords saved one's hours when typing at a genuine tty.

The advent of verylongvariablenames into programming brought the possibility of additional clarity. To avoid the difficulty inherent in reading runtogethernames, several solutions were proposed. The use of the underscore character, "\_", as a default substitute\_token\_for\_blank\_space, is tolerably satisfactory, as long as one is using terminals and displays that print the substitute^token^for^blank^space in a uniform manner, and as long as one is prepared to deal with the tendency for printed listings to tear just below lines that have many such delimited names. The use of the "\$" as the substitute\$token\$for\$blank\$space suggests greed, which is often correct but rarely appropriate. Of course, with proper capitalization, the substituteTokenForBlankSpace can be noTokenAtAll. Whatever the nature of the substitute.token.for.blank.space, the use of outrageously\_excessively\_long\_tokens\_which\_contain\_many\_instances\_of\_the\_substitute\_token\_for\_blank\_space, can cause the word-wrap function of many editors to experience noteworthy dismay.

In a coding style known as "Hungarian", variable names are prefixed with letter combinations that vindicate their types. For example, the prefix "n" might vindicate a noun, "v" a verb, "j" an adjective, and so forth.

((Lisp programmers) ((tend (to use)) (lots (of parentheses)) ((to indicate) structure.))) Forth used have who Programmers structure reverse Polish notation uniform use of by indicate. Begin Pascal programmers do begin tend to be extremely pedantic; and use lots of semicolons end while false ; end. {C programmers do {something similar; but more concisely;} while( 0 ) ;}

Choosing a style for commenting a FORTRAN program is rather simple. Complete adherence to the requirements of the language can be had merely by keeping handy a dictionary or

Crib sheet with a collection of useful words that begin with the letter  
C.  
C reativity is required, however, in  
Coping with paragraph indentation.

110110100011011011001100101010001111101011101100011011100101111010100  
01001101110110001011101011001110111011101101101101010111101001110. 198DF  
0CEB278467E239C87ED59287A598DF23798B3473D459EF8721AC08975E983D749BB837CA!

-- Jay Freeman