

What Kind of Programmer Are You?

by Norman Grabowsky

This test may tell you more than you want to know about your programming.

Here is a quick test which will enable you to evaluate yourself as a computer programmer. A fixed-point full word variable called *I* always has a value of one or two. Provide code to change the value of the variable *I*. If *I* is found to contain a value of one, change it to two and vice versa.

Find your category by first comparing your solution to each of the 10 solutions supplied, selecting the one that most closely resembles your solution, and choosing the category with the same number as the solution selected.

You may code in any language and compare your method with those shown below, but PL/1 is the choice we've used.

Solutions:

- (1) IF I = '2' THEN I = 1;
ELSE I = 2;
- (2) IF I = 2 THEN I = 1;
IF I = 1 THEN I = 2;
- (3) IF I = 1 THEN GO TO SKIP;
I = 1;
GO TO DONE;
SKIP: I = 2;
DONE:
- (4) J = 2;
IF I = 2 THEN J = 1;
I = J;
- (5) DECLARE SWITCH LABEL;
.
.
IF I = 1 THEN SWITCH = ONE;
IF I = 2 THEN SWITCH = TWO;
GO TO SWITCH;
ONE: I = 2;
GO TO DONE;
TWO: I = 1;
DONE:
- (6) DECLARE ONETWO(2) FIXED BIN(31)
INITIAL(2,1);
.
.
I = ONETWO(1);
- (7) I = 3 - I;
- (8) I = 1 - ((I/2 * 2) + 1);

- (9) IF I = 2
THEN DO;
I = 1;
END;
ELSE DO;
I = 2;
END;
- (10) IF I = 1 THEN I = 2;
ELSE DO;
IF 123 = 2 THEN PUT LIST
('BAD I—WILL SET TO 1');
I = 1;
END;

Categories

(1) *Recent IBM School Graduate*. I estimate not more than one out of ten recent IBM school graduates knows the difference between 2 and '2.' It is a mystery to me how anyone can successfully program without understanding such a basic concept.

(2) *Programming Instructor*. Like the preceding solution, this one is also incorrect. I question how many persons teaching programming have ever actually designed, coded, and debugged a program of any consequence.

(3) *Old-Line FORTRAN Programmer Who Just Finished a PL/1 Course*. There is a lot of truth to the saying: "You can't teach an old dog new tricks." This solution gives the proper results, but that is about all you can say for it.

(4) *Assembly Language Programmer Who Took the Same Course*. Actually this solution is quite reasonable for readability, but is not too straightforward and efficient. This is probably a result of PL/1 training. This solution is surprising in that it did not contain any self-modifying code, but then again this is most likely a result of returning to earth by learning a high level language.

(5) *Systems Analyst*. This best illustrates the school of thought that feels the duty of a systems analyst is to poorly design a program to such an extent that even a good programmer appears to be incompetent.

(6) *Computer Scientist*. This is probably the most clever of the solu-

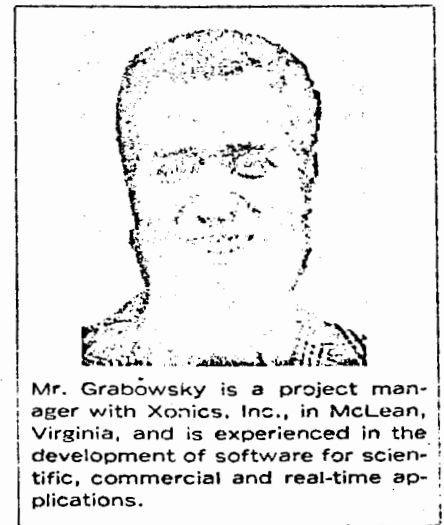
tions. However, it is not explained with a comment, and therefore mere mortals could have some problems comprehending it.

(7) *Mathematician*. This is very similar to the last solution in that it is also quite elegant but difficult to comprehend without further explanation.

(8) *Job Security-Minded Programmer*. This is quite a work of art and of course there are no comments. Would Macy's tell Gimbels? It actually works, but according to Lewis Carroll, "I could have thought of a much more complicated way of doing it said the red queen immensely proud."

(9) *Structured Programming Devotee*. One might say this is a cheap shot at structured programming. However, it seems to me that too many persons worry about trivialities as how many columns one should indent, rather than using the techniques as intended.

(10) *Good Programmer*. Notice that none of the other solutions checked *I* for out-of-range values. This is a very dangerous yet common practice. Unfortunately, all the structured programming, fancy algorithms, elegant solutions and efficient coding would not save you if for some reason *I* did not originally have a value of one or two. *



Mr. Grabowsky is a project manager with Xonics, Inc., in McLean, Virginia, and is experienced in the development of software for scientific, commercial and real-time applications.