**LETTERS TO THE EDITOR**

**Avast ye landlubber!**

**Editor:**
You folks may be “up” on computers, but you are not “up” on aircraft carriers.

In the computer-generated picture (Figure 4, p. 9, of “Computer Graphics: Reaching the User” by Ware Myers, March 1981 *Computer*), “the apparent twin sterns” are in fact “the apparent twin bows” of the depicted aircraft carrier. The part closest to the viewer is the front of the ship (bow), not the rear (stern).

The view presented in Figure 4 is of the flight deck as viewed from a plane that has just taken off (or is flying by) and looking back through a right-hand window over a pilot’s right-hand shoulder.

Hopefully, this view is not programmed into the stated “critical application” of “landing” an aircraft on an aircraft carrier...

Don Kaufman
T. R. Arnold & Associates
Elkhart, Indiana

**Author Ware Myers stands corrected.**

Ware claims to be a former Navy man, but we suspect his tour of duty consisted of personnel work at a training base in land-locked Idaho (years before the damming of the Snake and Columbia rivers entitled state boosters to claim Lewiston as a ‘seaport’). Ed.

**Ada profile one-sided?**

**Editor:**

I found your June issue, featuring the Ada language, to be of substantial interest, but I believe there is another side of the Ada story which was left out. I am referring to the comments of C. A. R. Hoare. In his acceptance speech for last year’s ACM Turing Award, Hoare made several remarks regarding what he feels to be the shortcomings of Ada. He considers the language to be unreliable, but did not go into detail at that time (I gather he has done so elsewhere).

As an IEEE-CS member, I feel that *Computer* should go to greater lengths to present both sides of the issue. Personally, Ada looks quite reliable to me, but I would like to see in greater detail what Hoare and (presumably) others are referring to. Perhaps an upcoming article might attempt to present their views.

Robert R. Lewis
Member of the Programming Staff
CALMA Co.

**Editor:**

This letter is protesting the presentation of the topic of Ada in the June 1981 *Computer*. The IEEE is an engineering society and as such should uphold reasonable standards of scientific and engineering presentations.

Although there have been many serious reservations about Ada expressed publicly by eminent people in the computer field, the entire presentation is exclusively one-sided. Computer science is struggling to be recognized as a science, and many of its students have only a slight appreciation of the scientific approach, so it is especially important in these times that the IEEE set high standards of careful presentation of various sides. Please note that this letter is neither for nor against Ada (though I have my own private opinions); it is against the kind of presentation you gave.

Richard W. Hamming
Naval Postgraduate School
Monterey, CA 93940


**Guest editor’s reply:**

It was our intention in the June special issue to present the Ada program to *Computer*’s readers. Like everything else, Ada has its pros and cons.

However, it is an undeniable fact that the program is happening and is almost certainly going to have an impact on all of us, no matter how we feel about it personally. We have chosen to simply present the facts about this major development so that readers could better form
need the genius of Shakespeare to use it effectively in speaking to a five-year-old child. On the other hand, Basic may be usable by someone with a limited mathematical background, but for anyone with any more mathematics than that, a language which admits expressions like "LET N = N + 1" makes needless difficulties.

F. G. Duncan
University of Bristol

Editor:
The correspondence regarding mnemonics in the May 1981 issue was very interesting—and a bit dismaying. It spurred me to reread the "Microprocessor Assembly Language Draft Standard," which is (as Mr. Duncan and Mr. Rogers suggest) extremely complex and non-extendable. More significantly, the response from Mr. Fischer, the Task P694 chairman, showed a total lack of comprehension of the suggestions raised by Mr. Duncan.

I myself have used an assembler for the Z80 in which (for example) the command to load register A from the contents of the address contained in register pair HL is not LD HL A, or LD HL A or LD A,HL but a = (HL). This is a lot closer to the RTL description of the operation, and is also easier to understand when users come from backgrounds of differing assemblers. While I was not previously aware of Mr. Duncan's work, I am sure that the developers of this Z80 assembler were.

Mr. Fischer apparently saw the "= " in Mr. Duncan's letter and decided that since it was clearly about Pascal he need not bother reading further. His response suggests that a machine-oriented subset of Pascal is being discussed, and that Basic would be better, when actually the discussion turns on upgrading assembler statement format to look more like Basic. An assembler of this form need not be significantly larger or slower than the "magic code" type and need not require any more (or any less) "operating system" support. (Mr. Fischer's mention of old assembly language code is also interesting—will an assembler for the standard mnemonics also attempt to take all possible old forms?)

In summation, I suggest that (1) Mr. Duncan's ideas be considered seriously by those seeking to standardize, as it is an algebraic-form common ground, and (2) Mr. Fischer be more careful to understand suggestions before disposing of them.

Harlan A. Rosenthal
Hackensack, New Jersey

To be considered for publication, a letter to the editor must be accompanied by a statement giving Computer permission to publish that letter.