May 29, 1981

M. Sabina H. Saib
Director, Software Quality Department
General Research Corporation
5383 Hollister Avenue
PO Box 6770
Santa Barbara, CA 93111

Dear M. Saib:

I am happy to accept your letter on Hoare's Turing Lecture for publication
in the ACM Forum. It is scheduled to appear in the July issue, along
with two other letters received and an author's response.

I have done some minor editing on your letter (copy attached). I did
take out the reference to the quicksort enclosures, because I think they
do not contribute enough to the general argument to warrant inclusion.

Thank you again for your interest in Communications.

                              Sincerely,

                              Robert L. Ashenhurst

RLA:dd
encl.
cc: C. A. R. Hoare
    J. Benton

May 29, 1981

Mr. Arthur Evans, Jr.
Bolt Beranek and Newman Inc.
10 Moulton Street
Cambridge, MA 02238

Dear Mr. Evans:

I am happy to accept your letter on Hoare's Turing Lecture for publication
in the ACM Forum. It is scheduled to appear in the July issue, along
with two other letters received and an author's response (containing the
points in his letter to you of 6th May, 1981).

Thank you again for your interest in Communications.

                              Sincerely,

                              Robert L. Ashenhurst

RLA:dd
cc: C. A. R. Hoare
     J. Benton

May 29, 1981

Mr. Daniel R. Hicks
513 Fourth Avenue NE
Box 815
Byron, MN 55920

Dear Mr. Hicks:

I am happy to accept your recent letter submitted to the ACM Forum for
publication. Your nicely tie in several topics from the February issue,
but I have elected to include it in a "package" of letters commenting on
Hoare's Turing Lecture. These will appear, along with an author's
response (which as regards your letter simply expresses agreement with
your point of view) in the July issue.

Thank you again for your interest in Communications.

                              Sincerely,

                              Robert L. Ashenhurst

RLA:dd
cc: C. A. R. Hoare
     P. J. Denning
     J. Benton

I sincerely hope that this study of the future will not come true for the profession of medicine or architecture or law or social work, or engineering. One reason is that I do not believe that all of professional practice can be codified in any set of books; nor do I believe that it is so far beyond the scope of human intellect and understanding that it needs such codification. Finally, I do not believe that any single manufacturer of supplies and equipment will have the temerity, let alone the commercial power, to construct and propagate the necessary catalogue and manuals.

omit.

But in the case of my own profession, that of computer science, I very much fear that it has already gone a long way on the path towards a priesthood. The analogies are too close for comfort. Our altars are hidden from the general public in a superbly air-conditioned holy of holies; ministered to day and night by a devoted team of acolytes, and regarded by the general public with feelings of mystical awe, which has been nourished in our own interests by those specialists who should have put forward all their efforts to dispel it.

These analogies are dangerous; but I believe that the most dangerous aspect of all is the increasing dominance of our sacred books , the *basic* software manuals which have become essential to our every approach to the computer, and whose size and complexity seems to be increasing at an *the same rate* unbelievable rate. I have recently attempted to study a slim volume *as our computers* from the library provided by one manufacturer. It was rather over a *are getting smaller* hundred pages of close type. At the beginning, it frankly told me that an understanding of the text would depend on prior familiarity with three other volumes not available to me. It also admitted that the practical application of the knowledge contained therein would depend on a knowledge of some other volumes, depending on the size and nature of my computer and its software configuration.

Nevertheless, I persevered in my study of this single volume. I found in it some extremely well written and well presented case studies of several example problems which could be solved with the aid of particular incantations displayed therein. Indeed, I had a very strong impression that I had understood the methods described, until I tried to apply them to a new and different example problem of my own. Then I realised that I had no knowledge of the actual capabilities of the software described - what was the range and limits of its adaptability. In fact, the only way of solving the new problem was to try to copy the incantations carefully from the manual, making as few adaptations as possible to meet the needs of the new problem. Naturally, this involved a great deal of guesswork, since there was no way of finding in advance whether the changes would have the desired effect, or indeed, whether they would be acceptable to the deity at all. The only recourse is to try them out; if successful, I will praise the lord from whom all good things do flow; if not, I will attribute my failure to my own errors, sins of commission and omission, and resolve to do better next time. Any idea that I have control or responsibility for the efficient and effective use of the computer has disappeared entirely.

software In the old days, our computers occupied racks and cabinets filling a complete room, which the programmer enters, carrying in his pocket a slim square of folded cardboard known as a "FACTS CARD". Nowadays the computer consists of our room is filled with shelves and cabinets laden with manuals, and we will soon enter it with our

There is another and rarer class of person whose activities are recorded by history, and whose successors are still active today - these are the prophets of doom. They are inclined to point out that the priesthood has gotten fat and idle and self-seeking; that they are leading the people in the worship of false gods, and that the only hope of avoiding disaster is a widespread return to the simple virtues of earlier times. Naturally, they make themselves very unpopular with the priesthood, and usually even more so with the population; it is not the priests but the population which drives such men and women into the wilderness.

In spite of this, I can see that prophets of doom are still needed today, and still persist in their unpopular messages. An outstanding example is Rachel Carson, who pointed out that the growing use of chemical insecticides is even more damaging to crops and people than the insects which they were increasingly failing to kill. And I hope that even in the future, if a profession ever degenerates into a priesthood, there will be those who continue to fight against it. In the future as in the past, we will have doctors who persist in pointing out that fashionable remedies are useless, or even damaging, that the drugs are worse than the disease, that many of them are addictive, that their effectiveness decreases with repeated use, that their side-effects induce even worse diseases, which need yet more drugs for their cure; until the problems caused by the drugs are many times more severe than the original problems which the drugs were designed to cure.

This situation has occurred again and again the history of medicine and there have always been doctors brave enough to point it out. It has occurred in the development of artificial fertilisers and weedkillers. And let us not be ashamed to admit that it has occurred in the development of computer software designs, which were originally intended to simplify computer usage and make it more accessible, and which have now become a major barrier to both understanding and access.

So it is among the prophets of doom/that I wish to enrol myself. I believe that the current situation in basic software design is bad, and that it is getting worse. I do not expect that this will be recognised by the designers, manufacturers, sellers, or even the users of software, who will regard the increase in complexity as a sign of progress, or at least an inevitable concomitant thereof; and may even welcome it as a tribute to their intelligence, or at least a challenge. The larger manufacturers are not only set in their ways, but they have actually profited from the increase in complexity of their software, and the resulting decrease in the efficiency and effectiveness of their customers' use of hardware. And everything is now so complicated, that any particular attack on the problem of low quality software design can always be evaded by appeals to tradition, to standards, to customer prejudice, to compensating advantages, and even to promises to mend the fault in future issues.

So it will be a long time before there is even any recognition of the problem which faces our profession. But even if the problem were widely recognised and deplored, its solution is going to present extremely

pressures make it impossible for the smaller companies to strike out on their own in a new and more promising direction. In fact, like everyone, basic they are forced to live with the same software pollution that affects us all, and

I suggest that if there is any hope for the future, it lies with the small computer manufacturers, and the software houses. They have a vested interest in cost/effectiveness, in the use of skill and professional qualities to achieve predictably good results at predictably low cost. But I fear that the weight of tradition and commercial

difficult technical problems. The pursuit of complexity is easy, and the implementation of complexity can safely be delegated to competent managers. But the pursuit of simplicity is one of the most difficult and challenging activities of the human mind. Progress is likely to be extremely slow, where each complexity eliminated must be hailed as a breakthrough. We need not only brilliance of intellect but breadth of experience, nicety of judgement, excellence of taste, and even more than our fair share of good luck. And finally we need a puritanical rejection of the temptations of features and facilities, and a passionate devotion to the principles of purity, simplicity and elegance.

And so my final message too has strong religious overtones. I hope that the reformation in software design that I advocate will be peaceful and prosperous.