(3) The predefined numeric types INTEGER and FLOAT were supported along with LONG_INTEGER; however, type FIXED was not available at all, nor could users define their own INTEGER or FLOAT types. Also, type conversion from LONG_INTEGER to FLOAT was not supported.

(4) One particularly vexing deficiency was that CONSTRAINT_ERROR was not raised when an attempt was made to GET an object outside the range for its type. For example:

```
subtype HUMAN_AGE is
  NATURAL range 1..150;
AGE:
HUMAN_AGE;
.
.
GET(AGE); --200 is entered, no
  error message
PUT(AGE); --200 is printed
```

(5) The TeleSoft compiler did not support subprogram calls by named parameter association, only by positional association. This was potentially confusing when there were a lot of formal parameters.

These were some of the problems we encountered in the beginning course. As mentioned earlier, they did not cause undue hardship. In the advanced course, however, features of Ada that set it apart from other languages and that make it such an exciting and powerful language were simply not available.

(1) Separate compilation is, of course, an enormous aid to software development. Unfortunately, TeleSoft Ada did not support separate compilation at all, except that a package and a program using the package (via a "with" clause) could be compiled separately. However, the package specification and the body could not be compiled separately. If one modifies the package (either the specification or the body), then not only must the package be recompiled, but the accompanying program as well.

(2) The use of generics to write truly general and versatile software can be easily appreciated by students; unfortunately, this major feature of Ada was not supported either.

(3) The package CALENDAR was not available.

(4) Tasks types and defined constants were not allowed, nor "delay" or "abort" statements.

(5) Variant records, tasking, and renaming were not fully supported.

(6) Subunits of compilation units were not implemented. And the list goes on.

The problems in using a compiler that lacks so many features of a language are obvious. Perhaps the worst was that users became uncertain about whether they had written correct Ada that the compiler hadn't accepted, or whether they had simply written incorrect Ada. Students were constantly asking themselves, "Is it me, or is it TeleSoft?"

It must be stated that TeleSoft Ada Version 1.3 has since been superseded by Version 1.5 and Version 2.1. As of this writing (January 1985), we have had very little experience with either of the newer versions, but so far there do not appear to be major differences between 1.5 and 1.3. It is hoped that 2.1 will correct many of the problems found in the earlier versions.

David Rudd
Computer Science Dept.
University of New Orleans

References

Leadership Roles in ARTIFICIAL INTELLIGENCE & SOFTWARE ENGINEERING RESEARCH

United Technologies Research Center, a recognized source for broad ranging research of the first rank, has gained wide recognition for significant achievements both in pure science and R&D. Here, you’ll discover a research environment tailored to your needs…top scientific researchers…ample funding…outstanding back-up services…a scientifically-oriented management…encouragement and rewards for publication of papers.

Now is the ideal time to join us as we accelerate our Computer Science research and application activities. If you’re ready for a new leadership role, consider one of the following opportunities to work on leading-edge technology:

Artificial Intelligence
You will investigate and define artificial intelligence-based approaches to problem solving in manufacturing, design, and diagnostic systems. You will provide technical/program leadership in defining/implementing systems for selected applications.

Requires a PhD in Computer Science with extensive experience in AI subfields, such as expert systems, natural language, processing and knowledge-based system implementation.

Software Engineering
You will provide technical and program leadership in research, synthesis and deployment of software engineering methodologies, Ada and Ada-support environments, and knowledge-based approaches to software development.

Requires PhD or equivalent in Computer Science with extensive experience in Ada, AI, software engineering, database/knowledge base management, operating systems, formal specifications and design language.

Find out more about your excellent career prospects that include outstanding benefit and compensation programs. Send your resume to: M. G. Marcin, United Technologies Research Center, Silver Lane, East Hartford, CT 06108.

U.S. Citizenship Required

An Equal Opportunity Employer