



THE UNIVERSITY OF ARIZONA  
TUCSON, ARIZONA 85721

DEPARTMENT OF COMPUTER SCIENCE

**Icon Newsletter #8**

Ralph E. Griswold

November 30, 1981

**1. Cg**

Tim Budd's language Cg, which adds a generator capability to C, is now available. This system uses Yacc as a preprocessor to convert Cg programs into C programs. A runtime support system handles the generation.

A technical report describing Cg and the system itself are available. See the request forms at the end of this document.

**2. Version 5 of Icon**

Version 5 of Icon, implemented in C, is now available (there is a form for requesting this system at the end of this Newsletter). This is a final version; no further language changes are anticipated. The Version 5 changes are:

- The reserved word syntax for string scanning, `scan e1 using e2`, has been replaced by the operator syntax `e1 ? e2`. There are no changes in the semantics of the scanning operation. The infix `?` operator has low precedence, just above `&`, and associates to the left.
- `transform e1 using e2`, `insert(s,i)`, and assignment to scanned substrings (such as `(move(1)) := "x"`) have been deleted from the language.
- Augmented assignment has been extended to include all infix operators except the assignment operators. For example, `s ?:= e1` scans `s` and replaces its value by the value of `e1`.
- Compound expressions may be grouped in parentheses and separated by commas. Such a construction is like a function call, except that no function is applied. This produces mutual goal-directed evaluation among the expressions. Thus `(e1, e2, . . . , en)` is equivalent to `e1 & e2 & . . . & en`.
- An integer-valued expression may be placed in front of a parenthesis-grouped list of expressions. In this case, the integer, `i`, selects the result of the `i`th expression in the list. For example, `2(e1, e2, . . . , en)` produces the result of `e2`. Negative integers select results relative to the right end of the expression list. Note that this construction is not the same as a case expression, since all expressions in the list are evaluated before one is selected.

Version 5 also contains some improvements in the implementation; it is significantly faster than Version 4, especially in string scanning. Thanks to help from Alan Segal at Cray Laboratories, Inc., this version of Icon can be adapted to run on PDP-11s that do not have floating-point processors.

There is a short document, *Version 5 of Icon*, a new reference manual, and a new implementation guide. See the document request form at the end of this Newsletter.

### 3. Other Implementation News

#### 3.1 Version 2.1 Implementation for PRIME Computers

Ed Feustel has completed an implementation of Version 2.1 for the PRIME 550, 650, and 750 computers. This implementation runs under the Georgia Tech Software Tools Subsystem. The Icon implementation will be distributed with the standard Georgia Tech Subsystem at no extra charge. Interested persons should contact

Dr. Phil Enslow  
School of Information and Computer Science  
Georgia Institute of Technology  
Atlanta, Georgia 30332

Persons who already have the Georgia Tech Subsystem may send tapes directly to

Dr. Edward Alvin Feustel  
Senior Research Consultant  
PRIME Computer Inc.  
500 Old Connecticut Path  
Framingham, Massachusetts 01701

#### 3.2 Icon for the ONYX C8002

John Polstra has completed conversion of Version 3 of the C implementation of Icon for the ONYX C8002 and is currently converting Version 4. A document by him that describes his experience with the conversion is available; see the document request form at the end of this Newsletter. The ONYX Icon implementation is available from him for the cost of materials and handling:

John D. Polstra  
Member of Technical Staff  
UNIDOT  
1026 West Maude, Suite 309  
Sunnyvale, California 94086

He has also successfully converted the C implementation of Icon to run on a PDP-11/34. While he is not in a position to distribute this system, he is willing to discuss the conversion process with interested persons.

### 4. Icon Book

A book on Icon is in process. This book, which will be published by Prentice-Hall, Inc., is designed to meet the need for documentation on Icon, which only commercial publication can do. It will not be a reference manual, but rather a description of Icon and how to use it.

The manuscript should be to the publisher by mid-spring. Additional information will appear in the next Newsletter.

### 5. Programming Corner

There are relatively few responses to the request for self-reproducing Icon programs, although a number of self-reproducing programs in other languages were offered. Viktors Berstis pointed out that the self-reproducing SNOBOL4 program given in Newsletter #7 could be shortened and made more easily usable by replacing OUTPUT by PUNCH. He also suggested the following program:

```
PUNCH = REWIND(5) INPUT;END
```

This solution is certainly shorter than the one given, although it is more of a 'self-copying' program than a self-reproducing one, since it relies on facilities of the operating system (including the capacity to rewind the standard input file).

The shortest self-reproducing Icon program was supplied by Steve Wampler (an inside job):

```
procedure main();x:="write(\\"procedure main();x:=\\" ,image(x));write(x);end"  
write("procedure main();x:=" ,image(x));write(x);end
```

## Request for Icon Documents

Please send the documents checked below to:

---

---

---

---

---

- An Implementation of Generators in C, TR 81-5*
- Icon for the ONYX C8002*
- Corrigenda for the Icon Version 4 Reference Manual*
- Version 5 of Icon*
- Reference Manual for the Icon Programming Language; Version 5, TR 81-4a*
- A Tour Through the C Implementation of Icon; Version 5, TR 81-11a*

Return this form to:

Ralph E. Griswold  
Department of Computer Science  
University Computer Center  
The University of Arizona  
Tucson, Arizona 85721  
USA

### Request for Cg/Version 5 Icon

*Note:* Cg and Version 5 of Icon are designed to run under Version 7 of UNIX on PDP-11 computers. Icon, as distributed, requires separate I and D spaces. If both Cg and Icon are requested, they are sent on the same tape.

Please send the programs checked below:

- Cg
- Icon Version 5

Contact Information:

name: \_\_\_\_\_

address: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

telephone: \_\_\_\_\_

cable/telex: \_\_\_\_\_

Magnetic Tape Information: All magnetic tapes are written in 9-track format.

Please specify your preferred tape recording characteristics:

- 1600 bpi
- 800 bpi
- tar* format
- tp* format

Return this form to:

Ralph E. Griswold  
Department of Computer Science  
University Computer Center  
The University of Arizona  
Tucson, Arizona 85721  
USA

Include either a magnetic tape or a check for \$15.00 payable to the University of Arizona.