

Save the Envelope! See the Lead Article Below

The Icon Newsletter

No. 38 – March 1, 1992



Contents

Newsletter Subscriptions ...	1
Version 8.5 of Icon ...	1
New Macintosh Icon ...	2
More on UNIX Icode Files ...	2
ICEBOL6 ...	3
Icon Applications ...	3
RBBS at the Icon Project ...	4
A Puzzle ...	4
Producing the Newsletter ...	5
ISIcon Release 1.0 ...	6
Icon Program Library Updates ...	6
ProIcon Spring Sale ...	6
Graphic Credits ...	6
From Our Mail ...	7
Ordering Icon Material ...	8
Newsletter Subscription Renewal Form ...	11

Newsletter Subscriptions

Presently the *Icon Newsletter* is sent, free of charge, to more than 5,200 subscribers. *Newsletter* costs have gone up considerably in recent years, largely because of a substantial increase in the number of subscribers. It now costs several thousand dollars just to print and mail each issue.

We have managed to keep the *Newsletter* on a non-paid basis, but doing so has become increasingly difficult. It may be necessary to institute a subscription fee

at some point, but for now we're just asking subscribers to confirm their interest in receiving the *Newsletter*. This will enable us to remove persons no longer interested in *Icon* and hence reduce our costs.

To continue receiving the *Newsletter* you *must* return the form on page 11. You can cut the address from the envelope in which this *Newsletter* was mailed and tape it in the space provided. Be sure, however, to note any changes in your address. You can return this form by postal mail or fax, or send equivalent information by electronic mail (see the publication box on page 7). *Exception:* If you subscribe to the *Icon Analyst*, updates to *Icon* source code, or updates to the *Icon* program library, you need not respond. We'll continue to send you the *Newsletter* as long as you maintain another paid subscription.

Version 8.5 of Icon

In recent months we've made substantial changes to the implementation of *Icon* to accommodate the *Icon* interpreter and compiler in a single system. We've also added X-Window capabilities for platforms that support X. Along the way, we've also added a few new features to *Icon* and made some improvements to the implementation that will make *Icon* more useful in some environments. The net result is Version 8.5.

It's taken us quite a while to get this far and we're not quite ready yet to distribute the compiler or the X-

Icon
Icon

Window facilities. But we do have the Version 8.5 interpreter available for MS-DOS (see the order form on page 10). Implementations of Version 8.5 for other platforms are in progress and will be announced in future *Newsletters* when they are ready.

What's New?

Version 8.5 has the following new functions:

- `chdir(s)`, which changes the current directory to `s`.
- `flush(f)`, which flushes the input/output buffers associated with `f`.
- `function()`, which generates the names of the Icon (built-in) functions.
- `sortf(X, i)`, which sorts the elements of the list or set `X`, but sorts list and record values by their `i`th field.

Version 8.5 also has several new keywords, including one that gives the total amount of memory allocated and keywords for the frequently used mathematical constants e , ϕ , and π .

The most significant implementation improvement that will be evident to users of Version 8.5 is its ability to allocate multiple storage regions. MS-DOS users who have problems with running out of memory with the former single 65KB string and block regions stand to benefit most from this change. On a typical MS-DOS platform, for example, Version 8.5 provides access to at least twice as much memory as Version 8.0.

Finally, for MS-DOS users, the Icon executor, `iconx`, now can find `icode(.icx)` files at any place on the PATH specification, making Icon programs much easier to use.

Source Code

The same source code is used for all implementations of Icon distributed by the Icon Project. Conditional compilation and defined constants are used to configure the source code for different operating systems and specific platforms.

Version 8.5 of the source code presently is available only in MS-DOS LHarc archive format; other formats will be released later. The MS-DOS source distribution contains configuration information for Borland C++, Intel Code Builder, Metaware High C, Microsoft C, Turbo C, and Zortech C++. An MS-DOS executable for the LHarc utility is included.

Persons who have computers that can read MS-DOS diskettes and handle LHarc format can use the MS-DOS source distribution for implementations of Version 8.5 of Icon on other platforms.

Note: Subscribers to Icon source updates get Version 8.5 source automatically.

New Implementation of Icon for the Macintosh

Don Klett has produced a stand-alone implementation of Icon for the Macintosh and placed it in the public domain.

This implementation is now being distributed by the Icon Project. Unlike the former version (which still is available), this new version does not require MPW to run.

The new version of Macintosh Icon was built using THINK C's console interface, not the standard Macintosh interface. The THINK C interface provides for entering a command line and selecting options such as input and output files via radio buttons and standard dialog boxes.

This new Macintosh implementation includes all the features for Version 8.0, but none of ProIcon's extensions.

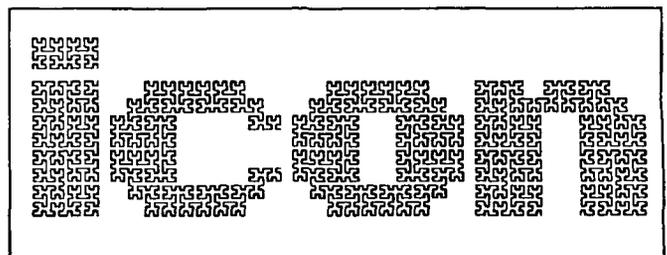
The source code also is available. It requires THINK C 5.0 and includes projects for THINK C.

See page 9 to order this new version of Macintosh Icon. Be sure to use the correct order codes: MET and MST for THINK C executables and source, as opposed to MEM and MSM for the MPW versions.

More on UNIX Icode Files

Richard Goerwitz sent us this comment on the article in the last Newsletter about making smaller icode files for UNIX platforms:

I just wanted to point out what you probably already know, namely that the BSD kernel hack that permits `#!/bin/sh` to work will not run under SYSVr3 or earlier. Some UNIXes, in fact, use the `#` to denote a C-Shell script (that is, if the first character in a file is `#`, the kernel executes it via the C Shell). Nearly all (all?) systems assume an un-adulterated shell script to be executed using the Bourne Shell, so it might be well to take out the `#!/bin/sh` from the specifications you give for reduced Icon headers.



ICEBOL6

ICEBOL6 will be held at Dakota State University, Madison, South Dakota, on October 15 - 16, 1992. The following information is taken from the conference announcement.

ICEBOL6, the Sixth International Conference on Symbolic and Logical Computing, is designed for teachers, scholars, and programmers who want to meet to exchange ideas about computer programming for non-numeric applications — especially those in the humanities. In addition to a focus on SNOBOL4, SPITBOL, and Icon, ICEBOL6 includes presentations on textual and logical processing in a variety of programming languages such as Prolog and C. Topics of discussion will include artificial intelligence and expert systems, and a wide range of analyses of texts in English and other natural languages. Parallel tracks of concurrent sessions are planned.

ICEBOL's coffee breaks, social hours, lunches, and banquet will provide a series of opportunities for participants to meet and informally exchange information.

Planned sessions include the following:

- analysis of texts (including bibliography, concordance, and index generation)
- artificial intelligence and expert systems
- computational linguistics
- computer languages and compilers designed for non-numeric processing
- electronic texts and encoding
- grammar and style checkers
- linguistic and lexical analysis (including parsing and machine translation)
- music analysis
- preparation of texts for publishing

Presentations at previous ICEBOL conferences were made by Paul Abrahams (former ACM President), Gene Amdahl (Andor Systems), Robert Dewar (New York University), Mark Emmer (Catspaw, Inc.), James Gimpel (Lehigh University), Ralph Griswold (The University of Arizona), Susan Hockey (Oxford University), Nancy Ide (Vassar College) and many others.

Copies of the ICEBOL5 Proceedings are available.

Requests for registration materials may be sent to:

Eric Johnson
ICEBOL Director
114 Beadle Hall
Dakota State University
Madison, SD 57042
U.S.A.

Inquiries and correspondence are encouraged via electronic mail, and they may be sent to Eric Johnson at:

eric@sdnet.bitnet

or

johnsone@dsuvax.dsu.edu

Icon Applications

We're sometimes asked what Icon is used for and, in particular, what major applications have been written in Icon. It's hard to define "major" in this respect. Some very small Icon programs are heavily used. And, of course, Icon programs tend to be shorter than, say, C programs for the same task. Certainly a 1,000-line program is large by Icon standards.

Despite the questionableness of size as a measure of significance, there is some correlation — large programs tend to have more functionality and, of course, they represent a significant investment by their authors.

We hear about uses of Icon, of course, but we have no way of knowing everything that's done with it. In hopes of getting more information than comes to us casually, a few months ago we sent a message to icon-group, the electronic news group for Icon, asking for information about large applications.

We received quite a few responses, despite the fact that icon-group reaches only a small percentage of the Icon user community. The amount of detail we received varied considerably, and a surprisingly large number of responses came from persons in commercial organizations who could not provide the programs for proprietary reasons. In fact, many persons didn't want their names or organizations mentioned.

What follows is a summary of responses, together with a few large applications with which we were already familiar. The descriptions are brief because in many cases we lack substantive information.

This list is far from complete, and it may not even be representative. Nonetheless, it shows the diversity of large Icon applications.

In this list, the organizational context for the application is indicated by letters as follows:

- A academic
- C commercial
- G governmental
- P personal
- R research (other than academic)

- C++-to-C translation (P)
- Computations in the Euclidean domain (A)
- Games (A and P)
- Biblical language retrieval (A)
- PL/I-to-C translation (R)
- Text formatting (A)
- Academic transcript evaluation (A)
- Report analysis (C)
- Infinite-precision base conversion (C)
- Symbolic Boolean expression manipulation (C)
- Hardware design and simulation (C)
- Test code generation (C)
- Information retrieval (G)
- PostScript generation (A)
- Assembly-language linker and library support (A)
- Voice processing (C)
- On-line order processing (A)
- Program analysis (A and C)
- Automated testing (C)
- Telephone system simulation (C)
- Object-oriented database manipulation (A)
- DNA analysis (R)

If you have a major Icon application in a category not listed here, we'd be interested in hearing about it.

We also appreciate copies of large programs. We use these for purposes such as analyzing the performance of the Icon compiler. Unless you specify otherwise, submitted programs will be for our own use only and will not be released to others. (Submissions for the Icon program library always are welcome.) 3.5" MS-DOS formatted diskettes are the easiest for us to process, but we can read most tape and diskette formats.

RBBS at the Icon Project

We've upgraded our remote bulletin board system at the Icon Project to support V.32 9600-baud transmission in addition to 2400- and 1200-baud transmission. The number is (602) 621-2283.

Most public-domain Icon programs are available from the RBBS. Some documentation also is available.

The RBBS normally is available from 5 p.m. to 8 a.m. Mountain Standard Time (midnight to 3 p.m. Greenwich Mean Time) on weekdays and all day on weekends and national holidays. It may be available at other times, but it is subject to unscheduled interruption then.

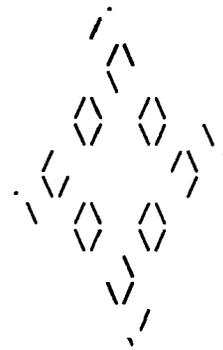
The RBBS is intended only for downloading. There are no provisions for uploading or answering messages.

A Puzzle

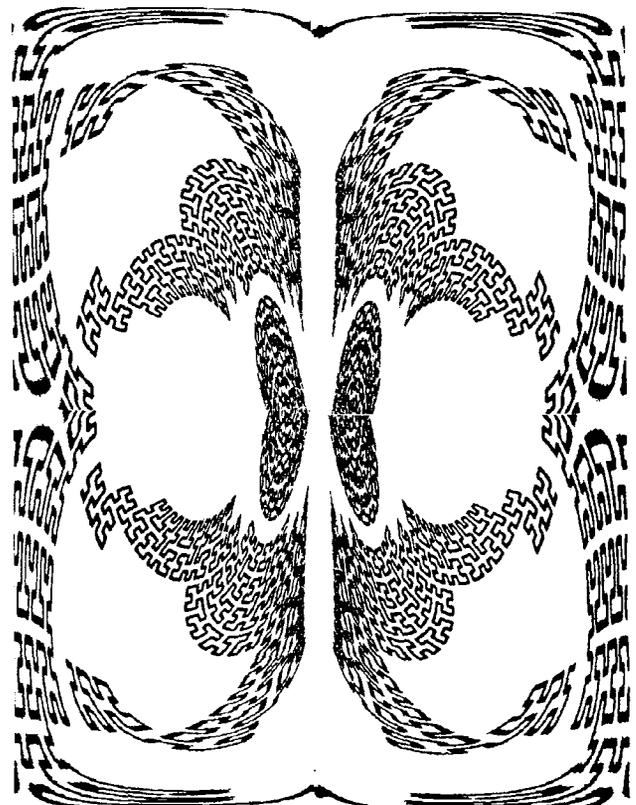
In the last *Newsletter*, we asked what the following characters represented:

```
2020 2020 2020 2e0a 2020 2020 202f 0a20
2020 2020 202f 5c0a 2020 2020 2020 5c0a
2020 2020 2f5c 2020 2f5c 0a20 2020 205c
2f20 205c 2f20 205c 0a20 202f 2020 2020
2020 202f 5c20 270a 2e20 5c2f 2020 2020
2020 202f 0a20 5c20 202f 5c20 202f 5c0a
2020 2020 5c2f 2020 5c2f 0a20 2020 2020
2020 5c0a 2020 2020 2020 5c2f 0a20 2020
2020 2020 202f 0a20 2020 2020 2020 270a
```

They are the hexadecimal codes for the ASCII characters that make up an Icon logo:



Our thanks to Gregg Townsend, the creator of the official Icon logo, for this one.



Producing the *Newsletter*

Getting together an issue of *Newsletter* is a lot of work, but it's also a lot of fun.

We do our own production. We use Aldus PageMaker 4.0 on a Mac IIfx for text entry, editing, composition, and layout. PageMaker is excellent for this kind of publication — its pasteboard paradigm is very intuitive, it has everything you need to produce a complete publication, and the story editor and spelling checker that come with PageMaker 4.0 make the production of a newsletter about as easy as we could imagine.

We print the *Newsletter* on a 300-dpi laser printer. We've tried producing photographic paper on an L-300 imagesetter at 1270 dpi, but the printed results don't look as good as the lower-resolution ones, probably because the printing process, which is not under our control, is not set up for high-resolution originals.

A few years ago, we started adding "clip art" to increase the visual interest of the *Newsletter*. Some of the artwork has been contributed and some we've done ourselves. We've used a lot of tools for composing and adapting artwork; recently we've relied primarily on Adobe Illustrator, Adobe Streamline, and Adobe Photoshop for this kind of activity.

We've wanted to add photographs of topical interest, and we've tried that a couple of times. The results have not been good, partly because the original photographs we've used have not been good and partly because we've generated camera-ready copy ourselves. We've also had problems getting the quality of printing we want.

We've produced camera-ready copy for photographs rather than going the more conventional route of letting the printer produce halftones from photographic prints. Our reasons are self-serving: We're interested in desktop publishing and all the intricacies involved, especially with artwork.

In *Newsletter 36*, we scanned 35mm photographic negatives using a Nikon LS-3500 scanner. This scanner is capable of extremely high resolution — much more than is needed for a newsletter like ours — but it's tricky to get images with the right contrast and gray levels. The scanned images are captured in Adobe Photoshop, where adjustments can be made, and then exported for use in PageMaker. In the case of photographs, we produce 2540-dpi images on the L-300 and then strip the results into the lower-resolution laser-printer output — a bit of a perversion and hardly a self-contained desktop-publishing solution.

The 2540-dpi images look good — nearly as good as photographic prints to the untrained eye. Getting good quality printing is another matter, but we're learning

what's needed the hard way and hope to have better-quality photographs in future *Newsletters*.

As we look back over the 14 years of the *Newsletter*, we can see truly immense improvements in its appearance. Part of the improvement is better technology for computer-assisted documentation, and part of it is what we've learned.

We hope to be able to continue to improve the appearance of the *Newsletter*. In the process, you'll be subjected to some experiments and some mistakes — we're presently printing 6,000 copies of each *Newsletter*, and we can't afford to have it done over because of minor problems in appearance.

Color printing has been in the back of our minds for some time. Spot color (one or two specific colors) wouldn't be difficult, nor would it be impossibly expensive. On the other hand, it wouldn't serve much purpose either. Our intention is not to produce eye-catching advertisements. But there have been times in the past when color would have served a really useful purpose. For example, our program-visualization tools use color in essential ways. It's bad enough to only be able to print snapshots of an animation, but we've been limited to black and white also. To do it right we need full color printing, not just a couple of spot colors.

For full color, what's needed is four-color-process printing, using what are called process colors: cyan, yellow, magenta, and black. There are four passes in the printing process, one for each color. Black is needed to compensate for the limitations of color inks in combination.

We can print process colors locally on a QMS ColorScript printer, which deposits hot wax on each of the four colors, rolling the paper back after each of the first three passes in order to print the subsequent one — a fascinating process to watch. It's a 300-dpi device, but with the color dithering used to get non-primary colors, its effective resolution is 75 dpi. The colors are reasonably good, but at 75-dpi, the images often are grainy. QMS output is fine for experimental work and disseminating things like color snapshots of a visualization to a small group. The printing process, however, is comparatively slow (about one minute per printed page) and too expensive, at about \$1 a page, for production work.

Just to give you an idea of what we'd like to be able to do in the way of color in the *Newsletter*, for a limited time we'll include a free QMS color snapshot of an Icon program visualization with any order of Icon material totalling more than \$25 or more. For example, a subscription to the *Icon Analyst* qualifies. Sorry — we can't provide these snapshots free or even sell them on an individual basis — the machine just isn't designed for mass printing. If you want a color snapshot and are

placing an order for \$25 or more, just write "snapshot" on your order form. To protect ourselves (and our QMS) this offer is only good until May 31, 1992. We'll consider extending the offer if the number of requests is manageable — and if our QMS hasn't gone south.

Iconic Software, Inc. Announces ISIcon Release 1.0

ISIcon Release 1.0, a basic version of the ISIcon programming language that contains all the features of ISIcon/SI except the Screen Interface functions, is now available for UNIX 386/486 and AT&T 3B2 computers. The new streamlined ISIcon interpreter is half the size of ISIcon/SI, but includes modules, function tracing, and the new readkb(), ab_exit() and isatty() built-in functions. All ISI products come with a 30-day money-back guarantee, 60 days of free technical support, and complete documentation. A single-CPU ISIcon license is priced at \$395 for UNIX 386/486 systems. Contact ISI for 3B2 pricing. There are no run-time royalties and ISIcon can be upgraded to ISIcon/SI for \$300.

ISI now offers a free ISIcon/SI demonstration disk. Contact them for details:

Iconic Software, Inc.
P.O. Box 3097
Lisle, IL 60532

voice: (708) 429-9166
fax: (708) 416-6772

uunet!isidev!isi



Icon Program Library Updates

There has been some misunderstanding about subscribing to updates to the Icon program library. The updates supplement the basic library, so subscribers to the library updates need the basic library also.

In cases where we have received new library update subscriptions from persons who did not order the basic library also, we have provided the basic library as a substitute for one update.

The current update subscription (updates 5 - 8) supplements Version 8.1 of the Icon program library.

Subscribers who started with update 1 have all the material in Version 8.1 of the Icon program library

If you have subscribed to the library updates recently and are uncertain about the status of your subscription, please contact the Icon Project.

ProIcon Spring Sale

The ProIcon Group is having a spring sale. For a limited time, Version 2.04 of ProIcon for the Macintosh, normally \$175, is available for only \$95. The sale ends May 31, 1992. No other discounts apply to the sale price. Add \$5 domestic, \$10 Canada, \$30 all other for shipping.

ProIcon features include

- a standard Macintosh interface
- an integrated editor
- functions for creating dialog boxes, manipulating windows and the clip board, navigating through folders, and performing other Macintosh operations
- access to HyperCard XFCNs and XCMDs
- royalty-free applications
- a separate application for visualizing storage management
- a comprehensive, 367-page manual

ProIcon is 32-bit clean and is compatible with Version 7.0 of the Macintosh operating system.

ProIcon can be ordered from:

Catspaw, Inc.

P.O. Box 1123
Salida, CO 81201-1123

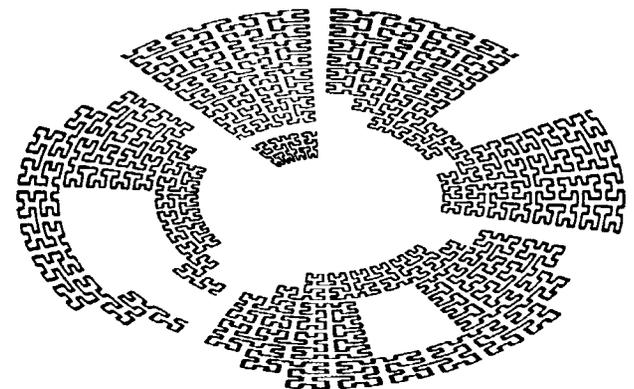
voice: (719) 539-3884
fax: (719) 539-4830



Graphic Credits

The graphic on page 2 was done by Lyle Rains, who commented "Since you are using graphics to fill odd spaces, what better than 'space-filling curves' to do the job?"

We produced the graphic on page 4 and the one below using Adobe PhotoShop to apply coordinate transformations to the graphic on page 2.



From Our Mail

Is there a proposal for an international standard for Icon?

We do not know of one. Note that since all existing implementations of Icon are based on the same source code, there is a substantial *de facto* standardization.



I found little joy in your recent "From Our Mail" column in which you indicated that you no longer have access to an Amiga with sufficient resources to compile Icon. I sincerely urge you to look into expanding the Amiga resources that are available to you so that those of us who use this wonderful platform may continue to have full access to your wonderful language.

Our problem is more complicated than just lacking an Amiga with sufficient resources to compile Icon. We don't had the personnel resources either. It's just not possible for the Icon Project to develop and maintain implementations of Icon for all possible platforms. In the past we have relied on others for help on nearly all the platforms for which Icon now is available. We'll continue to rely on this kind of support (several persons already have volunteered help for Amiga Icon). Such freely donated support is, however, understandably uncertain and sometimes unreliable. There often are delays in updates to Icon for platforms that we can't handle ourselves.

In a recent Newsletter, you mentioned a programming language SL5. What is it?

Ah, it's nostalgia time for us. SL5 was a programming language between SNOBOL4 and Icon. The name SL5 was an abbreviation for "SNOBOL Language 5". SL5 was an attempt to generalize and improve on the features of SNOBOL4. It featured a procedure mechanism in which the various aspects of procedure invocation, which normally are thought of as combined, could be specified separately. For example, creating a procedure environment, binding an argument list to it, and invoking it could be specified as separate operations in SL5. A general coroutine mechanism followed naturally

Downloading Icon Material

Most implementations of Icon are available for downloading electronically:

BBS: (602) 621-2283
FTP: cs.arizona.edu (cd /icon)

from this decomposition. Pattern matching in SL5 was done on "scanning environments", the pre-cursor of string scanning in Icon. There was a lot more to SL5, but that's history; SL5 is no more. The essential idea for Icon came out of SL5 with the realization that a full coroutine mechanism was not necessary for the search-and-backtracking control mechanisms needed by string scanning. In fact, generators in Icon are essentially a simplification of SL5's more general facilities.

Is there a version of Icon for Microsoft Windows in the works?

We had just written a reply that essentially said "no", when we got a call from Pete Roberts, who is planning on doing such an implementation. We hope to have more to report in the next *Newsletter*.

The Icon Newsletter

Madge T. Griswold and Ralph E. Griswold
Editors

The Icon Newsletter is published three times a year, at no cost to subscribers. To subscribe, contact

Icon Project
Department of Computer Science
Gould-Simpson Building
The University of Arizona
Tucson, Arizona 85721
U.S.A.

(602) 621-8448

fax: (602) 621-4246

Electronic mail may be sent to:

icon-project@cs.arizona.edu

or

...{uunet,allegra,noao}@arizona!icon-project

THE UNIVERSITY OF
ARIZONA
TUCSON ARIZONA

and



The Bright Forest Company
Tucson Arizona

© 1992 by Madge T. Griswold and Ralph E. Griswold
All rights reserved.

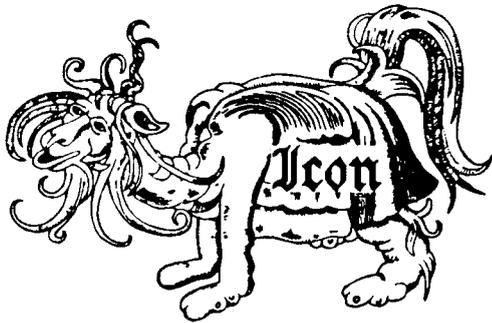
Ordering Icon Material

What's Available

There are implementations of Icon for several personal computers, as well as for CMS, MVS, UNIX, and VMS. Source code for all implementations is available. Program material is accompanied by installation instructions and users' manuals in printed and machine-readable form.

There also is a program library that contains a large collection of Icon programs and procedures, as well as an object-oriented version of Icon that is written in Icon.

In addition to users manuals that are included with program material, there are three books and two newsletters about Icon.



Icon Program Material

The current version of Icon is 8. Most program material listed on the next page is Version 8.0. See the note at the top of the left column on the next page concerning Version 8.5.

All program material listed on the next page is in the public domain.

Personal Computers: Executable files, source code, and the Icon program library for personal computers are provided in separate packages. Each package contains documentation in printed and machine-readable form. *Note:* Icon for personal computers requires at least 640KB of RAM; it requires more on some systems.

CMS and MVS: The CMS and MVS packages contain executable files, source code, test programs, the Icon program library, and documentation in printed and machine-readable form.

UNIX: The UNIX package contains source code (but not executable files), test programs, related software, the Icon program library, and documentation in printed and machine-readable form. UNIX Icon can be configured for most UNIX systems. *Note:* executable files for Xenix and the UNIX PC are available separately.

VMS: The VMS package contains object code, executable files, source code, test programs, the Icon program library, and documentation in printed and machine-readable form.

Porting: Icon source code for porting to other computers is distributed on MS-DOS format diskettes. There are two versions, one with a flat file system and one with a hierarchical file system.

Update Subscriptions: Updates to the Icon source code and the Icon program library are available by subscription.

Source-code updates are distributed on MS-DOS diskettes in LHarc format, and are suitable for compilation under MS-DOS and OS/2 or for porting to new computers. Each update usually provides a completely new copy of the source. A source-code subscription provides five updates. Updates are issued about three times a year.

Icon program library updates are distributed on MS-DOS diskettes in plain ASCII format. A library subscription provides four updates. Updates are issued about four times a year.

Documentation

In addition to the installation guides and users' manuals included with the program packages, there are three books on Icon. One contains a complete description of the language, the second describes the implementation of Icon in detail, and the third is an introductory text designed primarily for programmers in the Humanities.

There are two newsletters. *The Icon Newsletter* contains news articles, reports from readers, information of topical interest, and so forth. It is free, and is sent automatically to anyone who places an order for Icon material. There is a nominal charge for back issues of the *Newsletter*.

The Icon Analyst contains material of a more technical nature, including in-depth articles on programming in Icon. There is a subscription charge for the *Analyst*.

Payment

Payment should accompany orders and be made by check, money order, or credit card (Visa or MasterCard). The minimum credit card order is \$15. Remittance *must* be in U.S. dollars, payable to The University of Arizona, and drawn on a bank with a branch in the United States. Organizations that are unable to pre-pay orders may send purchase orders, subject to approval, but there is a \$5 charge for processing such orders.

Versions

The symbol \leftrightarrow identifies recently released material. The symbol \odot identifies Version 8.5 implementations. All others are 8.0.

Prices

The prices quoted here are good until May 31, 1992. After that, prices are subject to change without further notice. Contact the Icon Project for more current pricing information.

Ordering Instructions

Media: The following symbols are used to indicate different types of media:

- \odot 9-track magnetic tape
- \odot DC 300 XL/P cartridge
- \blacksquare 360K 5.25" diskette
- \square 400K 3.5" diskette
- \square 800K 3.5" diskette

All cartridges are written in raw mode. All diskettes are written in MS-DOS format except for the Amiga, the Atari ST, and the Macintosh.

CMS and MVS tapes are available only at 1600bpi. When ordering UNIX or VMS tapes, specify 1600 or 6250 bpi (1600 bpi is the default). When ordering diskettes that are available in more than one size, specify the size (5.25" is the default).

Shipping Charges: The prices listed include handling and shipping by parcel post in the United States, Canada, and Mexico. Shipment to other countries is made by air mail only, for which there are additional charges as noted in brackets following the price. For example, the notation \$15 [\$5] means the item costs \$15 and there is a \$5 shipping charge to countries other than the United States, Canada, and Mexico. UPS and express delivery are available at cost upon request.

Ordering Codes: When filling out the order form, use the codes given in the second column of the list to the right (for example, AME, ATE, ...).

Icon Executables

Amiga	AME	\square	\$15	[\$5]
Atari ST	ATE	\square	\$15	[\$5]
MS-DOS	DE	$\leftrightarrow \odot$ \square or \square	\$15	[\$5]
MS-DOS/386	DE-386	$\leftrightarrow \odot$ \square or \square	\$15	[\$5]
Macintosh	MET	\leftrightarrow \square	\$15	[\$5]
Macintosh/MPW	MEM	\square	\$15	[\$5]
OS/2	OE	\square or \square	\$15	[\$5]
UNIX PC	UE	\square or \square	\$15	[\$5]
Xenix	XE	\square (2) or \square	\$15	[\$5]
Xenix/386	XE-386	\square or \square	\$15	[\$5]

Icon Source

Amiga	AMS	\square	\$15	[\$5]
Atari ST	ATS	\square	\$15	[\$5]
MS-DOS; LHarc	DS	$\leftrightarrow \odot$ \square (3) or \square (2)	\$25	[\$5]
Macintosh	MST	\leftrightarrow \square	\$15	[\$5]
Macintosh/MPW	MSM	\square	\$15	[\$5]
Updates; LHarc (5)	SU	\square (2) or \square	\$60	[\$15]

Icon Program Library

Amiga	AML	\square	\$15	[\$5]
Atari ST	ATL	\square	\$15	[\$5]
MS-DOS and OS/2	DL	\square or \square	\$15	[\$5]
Macintosh	ML	\square	\$15	[\$5]
Porting; ASCII	PL	\square (2) or \square	\$15	[\$5]
UNIX; cpio	UL	\square (2) or \square	\$15	[\$5]
Updates; ASCII (4)	LU	\square (2) or \square	\$30	[\$12]

Complete Systems

CMS	CT	\odot	\$30	[\$10]
MVS	MT	\odot	\$30	[\$10]
UNIX; cpio	UT-C	\odot	\$30	[\$10]
UNIX; cpio	UC-C	\odot	\$45	[\$10]
UNIX; cpio	UD	\square (9) or \square (4)	\$40	[\$8]
UNIX; tar	UT-T	\odot	\$30	[\$10]
UNIX; tar	UC-T	\odot	\$45	[\$10]
VMS	VT	\odot	\$30	[\$10]

Books

<i>The Icon Programming Language</i> (2nd ed.)	LB	\$36	[\$13]
<i>The Implementation of Icon + update</i>	IB	\$50	[\$14]
<i>Icon Programming for Humanists + disk</i>	HB	\$33	[\$10]

Newsletters

<i>The Icon Newsletter</i> (all back issues, 1-37)	INC	\$17	[\$5]
<i>The Icon Newsletter</i> (back issues, each)	INS	\$1	[\$2*]
<i>The Icon Analyst</i> (1 year, 6 issues)	IA	\$25	[\$10]
<i>The Icon Analyst</i> (back issues, each)	IAS	\$5	[\$2*]

* Per order, regardless of the number of issues purchased.

Order Form

Icon Project • Department of Computer Science
Gould-Simpson Building • The University of Arizona • Tucson AZ 85721 U.S.A.

Ordering information: (602) 621-8448 • Fax: (602) 621-4246

name _____

address _____

city _____ state _____ zipcode _____

(country) _____ telephone _____

check if this is a new address

qty.	code	description	price	shipping*	total

	subtotal	
Make checks payable to The University of Arizona	sales tax (Arizona residents)	
	extra shipping charges*	
The sales tax for residents of the city of Tucson is 7%.	purchase-order processing	
It is 5% for all other residents of Arizona.	other charges	
Payment <input type="checkbox"/> Visa <input type="checkbox"/> MasterCard <input type="checkbox"/> check or money order	total	

I hereby authorize the billing of the above order to my credit card: (\$15 minimum)

card number

exp. date



name on card (please print) _____

signature _____



*Shipping charges apply only to addresses outside the United States, Canada, and Mexico

Newsletter Subscription Renewal

Please continue my free subscription to the *Icon Newsletter*. Use the following address:

I use Icon on the following platforms:

My main applications are:

I'd like to see more material of the following kind in the *Newsletter*:

Send this form to:

Icon Project
Gould-Simpson Building
Department of Computer Science
The University of Arizona
Tucson, AZ 85721
U.S.A.
fax: (602)-621-4246

Icon Project
Gould-Simpson Building
Department of Computer Science
The University of Arizona
Tucson, AZ 85721
U.S.A.
