# Installation and Maintenance Guide for Version 5.10 of Icon*

*Ralph E. Griswold*

*William H. Mitchell*

TR 85-15a

August 31, 1985; Revised October 14, 1985

Department of Computer Science

The University of Arizona

Tucson, Arizona 85721

*This work was supported by the National Science Foundation under Grant DCR-8401831.

**Installation and Maintenance Guide**
**for Version 5.10 of Icon**

Version 5.10 of Icon can be configured for the following computers running the UNIX* operating system:

AT&T 3B20S
IBM PC (PC/IX)
PDP-11 (models with separate instruction and data spaces)
Ridge 32
Sun Workstation
AT&T UNIX-PC
VAX-11

The distribution of Version 5.10 contains source code for Icon itself [1,2], documentation, sample programs and tests, the Icon program library [3], the procedures from the Icon book [1], test suites to aid in porting Icon [4], and various support material.

The installation procedure for Icon is simple; it requires unloading the distribution files onto the target machine, the setting of site-specific constants, and the compilation of the Icon system itself. Read through Sections 1 and 2 of this document before beginning the installation process.

## 1. Installation Information

### 1.1 Disk Storage Requirements

When the distribution files are unloaded, they occupies about 3.3 megabytes of disk space. During compilation, about 4.1 megabytes are required. To install subsidiary components and run all tests, about 5.5 megabytes are required. After the removal of unnecessary binary files, object files, and test results, about 4.6 megabytes are required. These figures vary slightly depending upon the logical organization of a particular file system.

More space can be saved by omitting machine-dependent portions of the system for computers that are not of interest, the Icon program library, source code for documentation, test programs, and so on. On systems with severely limited disk space, source code also can be deleted after Icon is installed. The executable binary files needed to run Icon occupy less that 150 kilobytes of disk space. See Section 3.2 and Appendices D and E.

### 1.2 Selecting the Directory for the Icon System

The distributed Icon hierarchy is rooted at v5. The default location for the Icon run-time system is v5/bin/iconx. This location is particularly important, since programs produced by the Icon translator reference the run-time system by its full path, and once the translator is built, iconx cannot be moved. *Important Note:* If a previous version of Icon is in use, read Section 3.1 before continuing.

Some care should be taken in the selection of the directory into which to unload the distribution files, since the default location of the executable binary files for Icon is subordinate to this directory. While other locations can be specified for these binary files, this involves additional work. Furthermore, on 4.*n*bsd UNIX systems, installation is complicated if the full path name to iconx is longer than 29 characters. The command processor that is needed to translate and run Icon programs, however, can be installed at any convenient location. See Appendices B and C for more detailed discussions of this matter.

File names used in subsequent sections usually are relative to the root directory for the Icon hierarchy. For example, if the Icon system is unloaded as described above, the root directory is

*UNIX is a trademark of AT&T Bell Laboratories.

/usr/icon/v5

and the file name v5/Makefile refers to

/usr/icon/v5/Makefile

Unless otherwise stated, the commands listed for the installation of Icon are to be done in v5.

### 1.3 Unloading the Distribution Files

The Icon distribution files are distributed in a variety of ways. The most frequent form of distribution is magnetic tape, although they it is available on 5-¼" diskettes in MS-DOS and PC/IX formats. The sections that follow refer to magnetic tape distribution only.

The Icon system is provided on tape in *tar* or *cpio* format, recorded at 800, 1600, or 6250 bpi as specified on the request form. Tapes are written in *tar* format at 1600 bpi if no specification is given. The format and recording density are marked on the label on the tape.

To unload the tape, do a *cd* to the directory that is to hold the Icon hierarchy (that is, the directory in which v5 is to be created) and mount the tape. The precise *tar* or *cpio* command to unload the distribution tape depends on the local environment. On a VAX running 4.*n*bsd, the following command should extract the contents of a 1600 bpi *tar* distribution tape:

```
tar x
```

On a PDP-11 running Version 7, the following command should work for such a tape:

```
tar xfb /dev/rmt0 20
```

Similarly, on a VAX running System V with a 6250 bpi tape, the following command should do:

```
cpio -icdB </dev/rmt/0h
```

As an example, if the Icon system is to reside at /usr/src/icon/v5 on a VAX running 4.2bsd, and the distribution tape is 1600 bpi in *tar* format, the following commands could be used to unload the tape:

```
cd /usr/src/icon
tar x
```

## 2. Installing Icon

Icon is installed using v5/Makefile. The following sections describe the process. Appendix A summarizes the makefile entries.

### 2.1 Configuring the Icon System

The configuration of the Icon system is done by the shell script v5/Icon-setup. Icon-setup accepts a number of parameters and modifies several source files to produce a ready-to-compile Icon system tailored as specified by the parameters. Icon-setup can be run a number of times; only the last run has any lasting effect.

There are entries in v5/Makefile for running Icon-setup with the parameters that are needed on most systems. If no special site-specific configuration is needed, one of these entries can be used to run Icon-setup:

| | |
|---|---|
| make Setup-att3b20 | AT&T 3B20 running System V |
| make Setup-pcix | IBM PC under PC/IX |
| make Setup-pdp11 | PDP-11 running Version 7 |
| make Setup-ridge | Ridge 32 running ROS 3.2 |
| make Setup-sun | Sun Workstation running 4.2bsd |
| make Setup-unixpc | AT&T UNIX-PC |
| make Setup-vax.bsd | VAX-11 running 4.*n*bsd |
| make Setup-vax.v | VAX-11 running System V |

If the parameters specified by these entries are not satisfactory, the setup can be done manually or the

v5/Makefile can be modified. The most likely need for a specialized setup is if iconx is to reside at a location that is different from the default one. See Appendix B for a complete list of setup options.

### 2.2 Compiling and Installing Icon

The Icon distribution tape contains no executable binary files and no object files, so the system must be completely recompiled from the source. After performing the setup, compile Icon with

```
make Icon
```

Compilation of the entire Icon system is a time-consuming process. For example, it takes about 9 cpu minutes on a VAX-11/785 running 4.2bsd, and is likely to take about 30 minutes of real time on a lightly loaded system.

On some systems, there may be a few warning messages during compilation. These should be ignored unless the resulting binary files prove to be unfunctional. A fatal error during recompilation is, of course, an indication of a serious problem.

After the Icon system has been compiled, the executable binaries are copied into place by

```
make Install
```

If the directory chosen for Icon binaries is not a public one, a copy of icont or a link to it may be put in a public directory.

The manual page icont.1 in v5/docs may be copied into /usr/man/man1.

### 2.3 Testing Icon

To test the newly compiled Icon system, use

```
make Samples
```

which runs a number of sample programs in v5/samples and compares their output with results from a VAX-11 running 4.2bsd at the University of Arizona. Some differences will show up in the test hello as a result of information that is date- and site-dependent. These differences are obvious and should not be cause for concern. Any other differences indicate a problem.

Although a successful run of these sample programs does not assure that Icon is functioning properly, any major installation problems should show up at this point.

There are many other test programs in the Icon system. Although it usually is unnecessary to run these before installing Icon for public use, they are available by

```
make Testtest
```

and

```
make Porttest
```

See [4] for more information. A list of differences that may be encountered when comparing results from these tests with the distributed results is contained in Appendix F.

### 2.4 Personalized Interpreters

Version 5.10 contains a facility for building personalized interpreters for Icon. This facility allows individuals to augment or modify the Icon run-time system easily and quickly. See [5] for details.

To install the personalized interpreter mechanism,

```
make Pi
```

The personalized interpreter mechanism can be tested by

```
make Pitest
```

This builds a sample personalized interpreter and runs tests, comparing the results from those from the

distribution tape. See Appendix F for a list of differences that may be encountered.

### 2.5 The Icon Program Library

The Icon program library contains a variety of Icon programs and procedures [3]. These may be installed for public use by

```
make Library
```

The library programs and procedures may be tested by

```
make Libtest
```

As with other test programs, the local output is compared with distributed output. See Appendix F for a list of differences that may be encountered.

## 3. Maintenance Information

The following sections contain information that is not needed to install Icon, but which may be helpful in understanding the organization of the system and in maintaining it.

### 3.1 The Effect of Version Changes on Interpretable Files

An interpretable file produced by icont contains a path to the interpreter, iconx. Thus, iconx cannot be moved without invalidating existing interpretable files. Furthermore, the interpreter for Version 5.10 is incompatible with those for previous versions. If, for example, Version 5.9 has been in use at a site and Version 5.10 is installed with the new iconx at the same location as that for Version 5.9, all previous interpretable files will be invalidated. To avoid this, it may be desirable to retain iconx for Version 5.9 at its present location and to put iconx for Version 5.10 at a new location, such as /usr/lib/icon/5.10/iconx. See the -iconx option in Appendix B.

### 3.2 Disk Utilization

As mentioned earlier, not all of the directories on the distribution tape are needed in order to install Icon. Once Icon is working satisfactorily,

```
make Clean
```

can be used to remove non-source files and test results. Additional disk space can be saved by deleting source code after the Icon system is built. See Appendices D, E, and G for more information.

### 3.3 Recompilation of System Components

There is a Makefile in v5 for carrying out various tasks, as described in preceding sections. This Makefile typically performs corresponding *makes* in subdirectories.

The subdirectories v5/src/iconx, v5/src/tran, and v5/src/link each contain code for a single component of Icon. Doing a *make* in any of these directories causes the particular component to be remade. The resulting component then can be installed by

```
make Install
```

in v5.

The subdirectories v5/src/fncs, v5/src/lib, v5/src/ops, v5/src/rt, and v5/src/iconx each contain source code for a part of the Icon run-time system. The Icon interpreter, iconx, is formed by linking all the run-time subroutines together with the routines in v5/src/iconx. When changes are made to the run-time system, all affected libraries must be rebuilt and then iconx must be rebuilt. For example, if the files v5/src/ops/bang.c and v5/src/fncs/read.c have been modified, the following sequence of commands rebuilds the system.

```
cd v5/src/fncs
make
cd ../ops
make
cd ../iconx
make
cd ../..
make Install
```

Alternatively,

```
make Icon
make Install
```

in v5 has the same affect.

### 3.4 Obtaining Source Code Listings

Execution of the command

```
make Listall
```

produces listings of all source files for the Icon system proper on standard output. Use the command

```
make List
```

to obtain listings of all such files that have been altered since the last make List or make Listall.

### 3.5 PDP-11 Yacc Modifications for the Icon Translator

This section is relevant only if modifications are to be made to the Icon grammar, which is contained in the file tran/icon.g. The version of Yacc distributed with VAX systems is large enough to build the Icon parser, but it may be necessary to build a version of Yacc with larger parameters on a PDP-11. The following defined constants in the file dextern (in the yacc source directory) should be given the values listed below. Larger values are acceptable for all these constants, but are not necessary.

```
# ifdef HUGE
# define ACTSIZE 3000
# define MEMSIZE 6000
# define NSTATES 300
# define NTERMS 127
# define NPROD 200
# define NNONTERM 100
# define TEMPSIZE 1200
# define CNAMSZ 4100
# define LSETSIZE 200
# define WSETSIZE 200
# endif
```

The constant HUGE should be defined instead of MEDIUM at the end of the file yacc/files. Then Yacc should be rebuilt.

## 4. Electronic Mail and Problem Reporting

A mailbox has been established to facilitate communication with the Icon Project. Use the following addresses for electronic mail:

| | |
|---|---|
| icon-project%arizona@csnet-relay | (CSNET and ARPANET) |
| arizona!icon-project | (Usenet and uucpnet) |

The Icon Project currently has uucp connections established through noao, mcnc, ihnp4, and utah-cs.

If any problems are encountered with the Icon system, send electronic mail or telephone the Icon Project at 602-621-6613. If these forms of communication are not convenient, use the Trouble Report Forms supplied with the distribution package.

**Acknowledgements**

Owen R. Fonorow, Janalee O'Bagy, and Gregg Townsend made a number of contributions to the procedures for installing and maintaining Version 5.10 of Icon.

**References**

1. Griswold, Ralph E. and Madge T. Griswold. *The Icon Programming Language.* Prentice-Hall Inc., Englewood Cliffs, New Jersey. 1983.

2. Griswold, Ralph E. and William H. Mitchell. *Version 5.10 of Icon.* Technical report, Department of Computer Science, The University of Arizona. August 1985.

3. Griswold, Ralph E. *The Icon Program Library; Version 5.10.* Technical Report TR 85-18, Department of Computer Science, The University of Arizona. August 1985.

4. Mitchell, William H. *Porting the UNIX Implementation of Icon; Version 5.10.* Technical Report TR 85-20, Department of Computer Science, The University of Arizona. August 1985.

5. Griswold, Ralph E. and William H. Mitchell. *Personalized Interpreters for Icon; Version 5.10.* Technical Report TR 85-17, Department of Computer Science, The University of Arizona. August 1985.

## Appendix A — Installation Makefile Entries

The following makefile entries may be used in the installation of Icon. They normally are used in the order given.

| | |
|---|---|
| make Setup-*name* | configure the Icon system |
| make Icon | compile the Icon system |
| make Install | install the Icon system |
| make Samples | perform basic tests |
| make Pi | build personalized interpreter system |
| make Pitest | test personalized interpreter system |
| make Library | build the Icon program library |
| make Clean | remove unnecessary files |

## Appendix B — Icon-setup

As described in Section 2.4, the configuration of the Icon system is done by the shell script Icon-setup. Icon-setup accepts a number of parameters and modifies several source files to produce a ready-to-compile Icon system tailored as specified by the parameters. Icon-setup is like any other UNIX command and all of its arguments must be specified on one logical command line. Icon-setup has the following synopsis:

Icon-setup
-sys *name*
-host *string*
[-hz *rate*]
[-nofp]
[-interpex]
[-vfork]
[-usg]
[-ibin *directory for Icon binary files*]
[-iconx *directory for the Icon interpreter*]

The parameters have the following meanings:

-sys

The -sys parameter selects the computer for which Icon is to be installed. The following names are supported: att3b, pdp11, mc68000, ridge, and vax.

-host *string*

Icon has a keyword, &host, whose value should be the name of the host machine where the system is running. On some systems, notably 4.2bsd, System III, and System V, it is possible to determine the name of the system at run-time via a system call. On other systems, 4.1bsd for example, the file /usr/include/whoami.h contains the name of the host in a #define statement. On 4.2bsd, specify gethost for *string*. This causes the *gethostname*(2) function to be used. On System III, System V, or some other system that supports the *uname*(2) system call, specify uname for *string*. On a system with a /usr/include/whoami.h file that has a #define for sysname, then specify whoami for *string*. If none of these are available on your machine, or to give &host some value besides that of the machine name, specify an arbitrary string (quotes around it may be needed) for *string*, for example: -host UNIX.

-ibin *bin-directory*

The *bin-directory* contains the Icon translator (itran), the Icon linker (ilink), and a header file (iconx.hdr) that is used to get Icon into execution. The path name of this directory is built into icont, the program that controls the translation and execution of Icon programs. By default, the fully qualified name of v5/bin is used for -ibin. Specifying -ibin causes the specified directory to be used instead.

-iconx *interpreter-directory*

The *interpreter-directory* contains iconx, the Icon run-time system. The full path to iconx is known to icont and is built into executable programs produced by icont. By default, the *interpreter-directory* is the fully qualified name of v5/bin. If -ibin is specified, then *interpreter-directory* defaults to *bin-directory*. If the full path name to iconx is longer than 29 characters and -interpex is specified, Icon-setup objects. See Appendix C if this occurs.

-interpex

Specify this option on a 4.*n*bsd system. This option causes the use of a feature of the *exec*(2) system call to make interpretable files directly executable. Do *not* specify this option on other systems. See Appendix C for more information.

-vfork

Specify this option if the operating system supports the *vfork*(2) system call; this should be specified for 4.*n*bsd systems.

-usg

Specify this option for System III, System V, or versions of UNIX derived from them.

-hz *rate*

This parameter specifies the cycle rate of the electrical environment. The rate defaults to 60 Hz. -hz does not need to be specified in a 60-Hz electrical environment. If the rate is incorrect, the value of &time will be wrong.

-nofp

Specify this option on a PDP-11 that does not have floating-point hardware.

## Appendix C — Direct Execution of Interpretable Files

When an Icon program is processed by the translator and linker using the *icont* command, the result is a file containing opcodes and data in a format that the Icon interpreter understands. Rather than having the user "execute" this interpretable file by running the Icon interpreter with the file as an argument, the Icon system uses one of two methods to make the interpretable files appear to be directly executable.

In 4.1bsd and 4.2bsd systems, a feature of *exec*(2) system call can be used to enable the interpretable file produced by the linker to appear to be directly executable. When *exec* is called with a file to execute, it examines the first two characters of the file. If the first two characters are #!, *exec* assumes that the next argument on the line is the name of a program for which the file is to serve as input. The program then is executed with the named file (the file that is being "executed") as its argument.

An alternative method is used on systems whose *exec*(2) system call doesn't have this feature. An executable file is prepended to the data used by the interpreter. The executable portion of the file merely runs the Icon interpreter with the file itself and any supplied arguments as the arguments for the interpreter.

If -interpex is specified for Icon-setup, the former method is used, otherwise, the latter method is used. The first method is preferable in that the interpretable files are smaller and they start executing more quickly.

There is a potential complication in using the first method. The 4.1bsd and 4.2bsd *exec*(2) system calls impose a length limitation of 29 characters on the name of the program to be run. If the name exceeds 29 characters, execution of the interpretable file fails. For example, suppose the Icon interpreter (iconx) on a system is located at /usr/csc/local/icon/v5/bin/iconx. This path name is longer than 29 characters, and is thus unsuitable for inclusion in interpretable files. The length of the path to iconx is checked by Icon-setup and the path above would be rejected.

One way to solve the problem is to link /usr/csc/local/icon/v5/bin/iconx to /usr/local/iconx, and have interpretable files reference /usr/local/iconx. Two things need to be done to accomplish this. First, find a location where a copy of bin/iconx can be referenced with a fully qualified path name that is no more than 29 characters long. Second, when configuring the system using Icon-setup, specify the new location of iconx using the -iconx option. For example:

```
Icon-setup other arguments -iconx /usr/local
```

It is also possible to get around this problem by not specifying -interpex and having Icon prepend the executable header on interpretable files.

## Appendix D — Icon Hierarchy

| | | | |
|---|---|---|---|
| v5 | root of the Icon system (location may vary from site to site) | | |
| | /library | Icon program library | |
| | | /src | source code for Icon library programs |
| | | | /cmd source code for programs |
| | | | /lib source code for procedure libraries |
| | | /ibin | executable binaries for programs |
| | | /ilib | linkable code for procedure libraries |
| | | /libtest | Icon library test programs |
| | | /man | manual |
| | | | /man0 front matter |
| | | | /man1 application programs |
| | | | /man2 procedures |
| | | | /man3 C functions |
| | | | ... ... |
| | | | /man7 miscellaneous |
| | | | /man8 library maintenance |
| | | | /cat0 formatted front matter for manual |
| | | | /cat1 formatted pages for application programs |
| | | | ... ... |
| | /rtlib | code for building personalized interpreters | |
| | /docs | Icon documentation | |
| | /book | source code for procedures from the Icon book | |
| | /bin | executable binaries for Icon | |
| | /src | source code for the Icon system | |
| | | /tran | source code for the Icon translator |
| | | /link | source code for the Icon linker |
| | | /h | header files for the Icon system |
| | | /fncs | source code for built-in functions |
| | | /ops | source code for operators |
| | | /rt | source code for run-time support routines |
| | | /lib | source code for routines called by the Icon interpreter |
| | | /iconx | source code for the Icon interpreter |
| | | /icont | source code for the Icon command processor |
| | | /sys | source code for target machine |
| | | /proto | source code for prototype implementation |
| | | /att3b | source code for AT&T 3B implementation |
| | | /pcix | source code for IBM PC/IX implementation |
| | | /pdp11 | source code for PDP-11 implementation |
| | | /ridge | source code for Ridge 32 implementation |
| | | /mc68000 | source code for Sun Workstation implementation |
| | | /unixpc | source code for AT&T UNIX-PC implementation |
| | | /vax | source code for VAX implementation |
| | | /pifncs | source code for Icon library C functions |
| | /pidem | sample personalized interpreter | |
| | /samples | Icon installation test programs | |
| | /test | Icon test suite | |
| | /port | Icon porting test suite | |

## Appendix E — Disk Utilization

The following table shows the approximate amount of disk space needed for the Icon system. Phase 1 refers to building Icon proper. Phase 2 refers to the installation of subsidiary components and running all tests. This data was obtained on a VAX running 4.2bsd. All figures are in kilobytes.

| | as distributed | after phase 1 | after phase 2 | after clean up |
|---|---|---|---|---|
| bin | 3 | 149 | 149 | 149 |
| book | 192 | 192 | 192 | 192 |
| docs | 452 | 452 | 452 | 452 |
| library | 605 | 605 | 907 | 844 |
| pidem | 1 | 1 | 487 | 487 |
| port | 256 | 256 | 450 | 256 |
| rtlib | 3 | 3 | 278 | 278 |
| samples | 47 | 47 | 63 | 47 |
| src | 1509 | 2123 | 2123 | 1617 |
| test | 253 | 253 | 361 | 253 |
| total (including root) | 3344 | 4102 | 5478 | 4591 |

## Appendix F — Test Result Differences

All the tests supplied with Version 5.10 of Icon compare locally produced results with results obtained produced on a VAX running 4.2bsd. The locally produced results are placed in a directory **local** that is subordinate to the test directory and are compared with corresponding files in in **distr**. In most cases, there should be no differences. In some cases, there are inevitable differences because the test results depend on time-, date-, or site-dependent data. In a few cases, test results are machine-dependent.

The following sections contain all the differences that are likely to be encountered. Any other differences should be regarded with suspicion and investigated carefully.

### Samples

The program **hello** will show differences due to time-, date-, and site-dependent data.

On an implementation for which co-expressions are not implemented, the programs **parallel**, **pdco**, and **parallel** will terminate prematurely with error messages.

No other differences should be encountered.

### Testtest

The programs in this set of tests are divided into six categories:

- **std** These programs test standard features of Icon and should produce the same results on all implementations.
- **ext** These programs test extensions to Icon and should produce the same results on all implementations.
- **exp** These programs test co-expressions and will terminate prematurely on implementations for which co-expressions are not implemented.
- **org** These programs test features of Icon that exhibit different results for 16- and 32-bit computers because of differences in the organization of sets and tables. There should be no differences for 32-bit computers, but there will be extensive differences for 16-bit computers. Such differences do not, *per se*, indicate implementation problems.
- **mem** These programs terminate prematurely on computers with limited address spaces.
- **chk** The program **chk01** will show differences because of time-, date-, and site-dependent data. The program **chk02** may show small differences because of different handling of floating-point arithmetic on different systems. Such differences do not indicate, *per se*, an implementation problem. The program **chk03** may show differences because of input/output idiosyncrasies on different systems. Differences should be checked closely to determine if they represent implementation problems.

### Porttest

No differences should be encountered in these tests.

### Pitest

The program **getenv** will show differences because of site-dependent data. The program **iscope** will show differences, since its output depends on memory locations and architectural properties of the system on which it is run. The program **ttyctl** will terminate prematurely on systems that are not running 4.*n*bsd.

**Libtest**

The program **farb** normally will show a difference, since its output depends on the time of day when it is run.

The program **loadmap** will show extensive differences on any system other that a VAX running 4.2bsd, since its output depends on the computer architecture and system libraries.

On an implementation for which co-expressions are not implemented, the programs **gpack**, **pdae**, **pdco**, **seqimage**, and **worm** will terminate prematurely with error messages.

## Appendix G — Listing of Distributed Icon Files

Files names followed by a slash are directories. Asterisks identify executable files, which are shell scripts.

| | | | | |
|---|---|---|---|---|
| Icon-setup* | bin/ | icon-pi* | port/ | src/ |
| Makefile | book/ | library/ | rtlib/ | test/ |
| Pimakefile | docs/ | pidem/ | samples/ | |
| bin: | | | | |
| Makefile | | | | |
| book: | | | | |
| 01/ | 08/ | 13/ | 18/ | alpha.ls |
| 02/ | 09/ | 14/ | 19/ | f/ |
| 04/ | 10/ | 15/ | 20/ | page.ls |
| 05/ | 11/ | 16/ | Makefile | |
| 07/ | 12/ | 17/ | README | |
| book/01: | | | | |
| countm1.icn | hello1.icn | hello3.icn | hello5.icn | locate2.icn |
| countm2.icn | hello2.icn | hello4.icn | locate1.icn | |
| book/02: | | | | |
| break1.icn | break2.icn | break3.icn | next.icn | |
| book/04: | | | | |
| balop.icn | inset2.icn | minmax2.icn | vbars.icn | wordlist1.icn |
| icwrite.icn | lmark.icn | powers.icn | word1.icn | words1.icn |
| inset1.icn | minmax1.icn | section.icn | word2.icn | |
| book/05: | | | | |
| array.icn | get.icn | tabwords1.icn | wordlen.icn | wordlist2.icn |
| book/07: | | | | |
| exor1.icn | expr2.icn | fib2.icn | fword2.icn | |
| expr1.icn | fib1.icn | fword1.icn | nword.icn | |
| book/08: | | | | |
| maxel.icn | words2.icn | | | |
| book/09: | | | | |
| copy1.icn | copy2.icn | copy3.icn | | |
| book/10: | | | | |
| display.icn | shuffle1.icn | | | |
| book/11: | | | | |
| fibseq1.icn | mark.icn | rtl.icn | | |
| genword.icn | marker.icn | to.icn | | |
| book/12: | | | | |
| tabwords2.icn | words3.icn | | | |
| book/13: | | | | |
| alt.icn | equalseq.icn | every.icn | filter1.icn | inter.icn |
| book/14: | | | | |
| 8q.icn | cross1.icn | limit1.icn | limit3.icn | stars.icn |
| break4.icn | cross2.icn | limit2.icn | posint.icn | |
| book/15: | | | | |
| abc.icn | arbno.icn | parsexp.icn | recexp.icn | tab.icn |
| arb.icn | lmatch.icn | rarb.icn | shades.icn | |

book/16:

| | | | | |
|---|---|---|---|---|
| eq.icn | lgraph.icn | stree.icn | visit.icn | |
| ldag.icn | ltree.icn | teq.icn | | |

book/17:

| | | | | |
|---|---|---|---|---|
| close.icn | disp.icn | reverse.icn | shuffle2.icn | swap.icn |

book/18:

| | | | | |
|---|---|---|---|---|
| add1.icn | add2.icn | add3.icn | mpy.icn | |

book/19:

| | | | | |
|---|---|---|---|---|
| drv.icn | form1.icn | symadd.icn | | |
| fix.icn | form2.icn | symop.icn | | |

book/20:

| | | | | |
|---|---|---|---|---|
| rsg1.icn | | | | |

book/f:

| | | | | |
|---|---|---|---|---|
| 8qp.icn | both.icn | fact.icn | infix.icn | rotate.icn |
| abcd.icn | btree.icn | filerev.icn | large.icn | rsg2.icn |
| acker1.icn | cdigit.icn | filter2.icn | lastline.icn | rsg3.icn |
| acker2.icn | charimage.icn | filter3.icn | limit4.icn | seqimage.icn |
| ackertr.icn | complex.icn | first.icn | locate3.icn | space.icn |
| allbal1.icn | dashes.icn | fixfunc.icn | nchars.icn | symmpy.icn |
| allbal2.icn | delete1.icn | form3.icn | oddline.icn | symsub.icn |
| allbal3.icn | delete2.icn | gener.icn | palseq.icn | tabwords3.icn |
| aver.icn | depth.icn | genpos.icn | pause.icn | tcopy.icn |
| bincop.icn | enrepl.icn | gensubstr.icn | qseq.icn | uscore.icn |
| boldface.icn | exor2.icn | hexcvt.icn | repalt.icn | vcount.icn |

docs:

| | | | | |
|---|---|---|---|---|
| Makefile | icon-pi.1 | tr83-3 | tr85-18 | tr85-20 |
| README | icont.1 | tr85-15 | tr85-19 | tr85-20a.roff |
| cover | reportform | tr85-16 | tr85-19a.roff | tr85-20b.roff |
| distpack | tmac.tr | tr85-17 | tr85-19b.roff | tr85-20c.roff |

library:

| | | | | |
|---|---|---|---|---|
| Makefile | ilib/ | man/ | | |
| ibin/ | libtest/ | src/ | | |

library/ibin:

library/ilib:

library/libtest:

| | | | | |
|---|---|---|---|---|
| Functest* | t-complex.dat | t-gpack.dat | t-math.icn | t-shuffle.icn |
| Makefile | t-complex.icn | t-gpack.icn | t-parens | t-size.dat |
| Proctest* | t-cppp | t-groupsort | t-patterns.dat | t-size.icn |
| Prog1test* | t-cppp.dat | t-groupsort.dat | t-patterns.icn | t-snapshot.dat |
| Prog2test* | t-cross.dat | t-i-psort.dat | t-pdae.dat | t-snapshot.icn |
| README | t-csgen.dat | t-i-split | t-pdae.icn | t-structs.dat |
| distr/ | t-deal.dat | t-i-split.dat | t-pdco.dat | t-structs.icn |
| func.tlist | t-delam | t-i-trfil | t-pdco.icn | t-strutil.dat |
| local/ | t-delam.dat | t-i-trfil.dat | t-queens.dat | t-strutil.icn |
| pdef.h | t-delamc | t-i-xref.dat | t-radcon.dat | t-tablc.dat |
| proc.tlist | t-delamc.dat | t-image.dat | t-radcon.icn | t-tablw.dat |
| prog1.tlist | t-edscript.dat | t-image.icn | t-roffcmds.dat | t-trig.icn |
| prog2.tlist | t-escape.dat | t-iscope.icn | t-rsg.dat | t-trim.dat |
| sizes.c | t-escape.icn | t-labels.dat | t-seek.icn | t-ttyctl.icn |
| t-bitops.dat | t-farb.dat | t-lam | t-segment.dat | t-ttyinit.dat |
| t-bitops.icn | t-fset | t-lam.dat | t-segment.icn | t-ttyinit.icn |
| t-bold.dat | t-gcomp | t-ll.dat | t-seqimage.dat | t-worm |
| t-bold.icn | t-gener.dat | t-lmap.dat | t-seqimage.icn | t-zipsort.dat |
| t-collate.dat | t-gener.icn | t-lmap.icn | t-shuffile.dat | |
| t-collate.icn | t-getenv.icn | t-loadmap | t-shuffle.dat | |

library/libtest/distr:

| | | | | |
|---|---|---|---|---|
| bitops.out | farb.out | iscope.out | radcon.out | tablc.out |
| bold.out | fset.out | labels.out | roffcmds.out | tablw.out |
| collate.out | gcomp.out | lam.out | rsg.out | trig.out |
| complex.out | gener.out | ll.out | seek.out | trim.out |
| cppp.out | getenv.out | lmap.out | segment.out | ttyctl.out |
| cross.out | gpack.out | loadmap.out | seqimage.out | ttyinit.out |
| csgen.out | groupsort.out | math.out | shuffile.out | worm.out |
| deal.out | i-psort.out | parens.out | shuffle.out | zipsort.out |
| delam.out | i-split.out | patterns.out | size.out | |
| delamc.out | i-trfil.out | pdae.out | snapshot.out | |
| edscript.out | i-xref.out | pdco.out | structs.out | |
| escape.out | image.out | queens.out | strutil.out | |

library/libtest/local:

library/man:

| | | | | |
|---|---|---|---|---|
| Makefile | cat6/ | man1/ | man7/ | tmac.an6n |
| cat1/ | cat7/ | man2/ | man8/ | tmac.an6t |
| cat2/ | cat8/ | man3/ | tmac.an | tmac.ilib |
| cat3/ | man0/ | man6/ | tmac.an.new | |

library/man/cat1:

| | | | | |
|---|---|---|---|---|
| cppp.1 | fset.1 | i-trfil.1 | loadmap.1 | tablc.1 |
| csgen.1 | gcomp.1 | i-xref.1 | parens.1 | tablw.1 |
| delam.1 | groupsort.1 | labels.1 | roffcmds.1 | trim.1 |
| delamc.1 | i-psort.1 | lam.1 | rsg.1 | zipsort.1 |
| edscript.1 | i-split.1 | ll.1 | shuffile.1 | |

library/man/cat2:

| | | | | |
|---|---|---|---|---|
| bitops.2 | gener.2 | pdae.2 | shuffle.2 | ttyinit.2 |
| bold.2 | gpack.2 | pdco.2 | size.2 | |
| collate.2 | image.2 | radcon.2 | snapshot.2 | |
| complex.2 | lmap.2 | segment.2 | structs.2 | |
| escape.2 | patterns.2 | seqimage.2 | strutil.2 | |

library/man/cat3:

| | | | | |
|---|---|---|---|---|
| getenv.3 | math.3 | trig.3 | | |
| iscope.3 | seek.3 | ttyctl.3 | | |

library/man/cat6:

| | | | | |
|---|---|---|---|---|
| cross.6 | deal.6 | farb.6 | queens.6 | worm.6 |

library/man/cat7:

| | | | | |
|---|---|---|---|---|
| i-hier.7 | | | | |

library/man/cat8:

| | | | | |
|---|---|---|---|---|
| lman.8 | uman.8 | | | |

library/man/man0:

| | | | | | |
|---|---|---|---|---|---|
| ptx.in | toc.in | toc2 | toc4 | toc6 | toc8 |
| ptxx | toc1 | toc3 | toc5 | toc7 | |

library/man/man1:

| | | | | |
|---|---|---|---|---|
| cppp.1 | fset.1 | i-trfil.1 | loadmap.1 | tablc.1 |
| csgen.1 | gcomp.1 | i-xref.1 | parens.1 | tablw.1 |
| delam.1 | groupsort.1 | labels.1 | roffcmds.1 | trim.1 |
| delamc.1 | i-psort.1 | lam.1 | rsg.1 | zipsort.1 |
| edscript.1 | i-split.1 | ll.1 | shuffile.1 | |

library/man/man2:

| | | | | |
|---|---|---|---|---|
| bitops.2 | gener.2 | pdae.2 | shuffle.2 | ttyinit.2 |
| bold.2 | gpack.2 | pdco.2 | size.2 | |
| collate.2 | image.2 | radcon.2 | snapshot.2 | |
| complex.2 | lmap.2 | segment.2 | structs.2 | |
| escape.2 | patterns.2 | seqimage.2 | strutil.2 | |

library/man/man3:

| | | | | |
|---|---|---|---|---|
| getenv.3 | math.3 | trig.3 | | |
| iscope.3 | seek.3 | ttyctl.3 | | |

library/man/man6:

| | | | | |
|---|---|---|---|---|
| cross.6 | deal.6 | farb.6 | queens.6 | worm.6 |

library/man/man7:

i-hier.7

library/man/man8:

| | |
|---|---|
| lman.8 | uman.8 |

library/src:

| | | |
|---|---|---|
| Makefile | cmd/ | lib/ |

library/src/cmd:

| | | | | |
|---|---|---|---|---|
| Makefile | delamc.icn | i-psort.icn | ll.icn | shuffile.icn |
| cppp.icn | edscript.icn | i-split.icn | loadmap.icn | tablc.icn |
| cross.icn | farb.icn | i-trfil.icn | parens.icn | tablw.icn |
| csgen.icn | fset.icn | i-xref.icn | queens.icn | trim.icn |
| deal.icn | gcomp.icn | labels.icn | roffcmds.icn | worm.icn |
| delam.icn | groupsort.icn | lam.icn | rsg.icn | zipsort.icn |

library/src/lib:

| | | | | |
|---|---|---|---|---|
| Makefile | escape.icn | patterns.icn | seqimage.icn | strutil.icn |
| bitops.icn | gener.icn | pdae.icn | shuffle.icn | ttyinit.icn |
| bold.icn | gpack.icn | pdco.icn | size.icn | |
| collate.icn | image.icn | radcon.icn | snapshot.icn | |
| complex.icn | lmap.icn | segment.icn | structs.icn | |

pidem:

port:

| | | | | |
|---|---|---|---|---|
| Linkchecker* | basis4.icn | esusp1.icn | lsusp1.icn | roman.icn |
| Linktest* | basis5.icn | esusp2.icn | meander.icn | rsg.icn |
| Makefile | basis6.icn | fail.tlist | prefix.icn | seqimage.icn |
| Runtest* | basis7.icn | fail1.icn | pret.tlist | set1.tlist |
| Runtestall* | basis8.icn | fail2.icn | pret1.icn | suspend.tlist |
| Trantest* | btrees.icn | gc.tlist | pret2.icn | suspend1.icn |
| arith.tlist | cross.icn | gc1.icn | pret3.icn | suspend2.icn |
| arith1.icn | display.tlist | gc2.icn | proto.icn | wordcount.icn |
| basis.tlist | display1.icn | hello.icn | psusp.tlist | |
| basis1.icn | display2.icn | lit.icn | psusp1.icn | |
| basis2.icn | distr/ | local/ | psusp2.icn | |
| basis3.icn | esusp.tlist | lsusp.tlist | recogn.icn | |

port/distr:

| | | | | |
|---|---|---|---|---|
| arith1.out | cross.u2 | lit.u1 | proto.u1 | rsg.ux |
| basis1.out | cross.ux | lit.u2 | proto.u2 | seqimage.u1 |
| basis2.out | display1.out | lit.ux | proto.ux | seqimage.u2 |
| basis3.out | display2.out | lsusp1.out | psusp1.out | seqimage.ux |
| basis4.out | esusp1.out | meander.u1 | psusp2.out | suspend1.out |
| basis5.out | esusp2.out | meander.u2 | recogn.u1 | suspend2.out |
| basis6.out | fail1.out | meander.ux | recogn.u2 | wordcount.u1 |
| basis7.out | fail2.out | prefix.u1 | recogn.ux | wordcount.u2 |
| basis8.out | gc1.out | prefix.u2 | roman.u1 | wordcount.ux |
| btrees.u1 | gc2.out | prefix.ux | roman.u2 | |
| btrees.u2 | hello.u1 | pret1.out | roman.ux | |
| btrees.ux | hello.u2 | pret2.out | rsg.u1 | |
| cross.u1 | hello.ux | pret3.out | rsg.u2 | |

port/local:

rtlib:

| | |
|---|---|
| Makefile | Rtlib* |

samples

| | | | | |
|---|---|---|---|---|
| Makefile | diffwords icn | parallel dat | recogn icn | tlist |
| Test* | distr/ | parallel icn | roman dat | wordcount dat |
| btrees dat | hello dat | pdco dat | roman icn | wordcount icn |
| btrees icn | hello icn | pdco icn | seqimage dat | |
| cross dat | local/ | prefix dat | seqimage icn | |
| cross icn | meander dat | prefix icn | sieve dat | |
| diffwords dat | meander icn | recogn dat | sieve icn | |

samples/distr

| | | | | |
|---|---|---|---|---|
| btrees out | hello out | pdco out | roman out | wordcount out |
| cross out | meander out | prefix out | seqimage out | |
| diffwords out | parallel out | recogn out | sieve out | |

samples/local

src

| | | | | |
|---|---|---|---|---|
| Makefile | icont/ | mc68000/ | pifncs/ | sys/ |
| att3b/ | iconx/ | ops/ | proto/ | tran/ |
| fncs/ | lib/ | pcix/ | ridge/ | unixpc/ |
| h/ | link/ | pdp11/ | rt/ | vax/ |

src/att3b

| | | | | |
|---|---|---|---|---|
| Makefile | csv s | icont c | ldfps s | setbound s |
| Setup* | defs s | ilink c | lsusp s | special s |
| arith s | display c | init c | params h | start s |
| coact s | efail s | interp s | pfail s | suspend s |
| cofail s | esusp s | invoke s | pret s | sweep c |
| coret s | fail s | ixhdr c | psusp s | |
| create c | gcollect s | lcode c | refresh c | |

src/fncs

| | | | | |
|---|---|---|---|---|
| Makefile | exit c | member c | reads c | system c |
| abs c | find c | move c | real c | tab c |
| any c | get c | numeric c | repl c | table c |
| bal c | image c | open c | reverse c | trim c |
| center c | insert c | pop c | right c | type c |
| close c | integer c | pos c | seq c | upto c |
| collect c | left c | proc c | set c | write c |
| copy c | list c | pull c | sort c | writes c |
| cset c | many c | push c | stop c | |
| delete c | map c | put c | string c | |
| display c | match c | read c | sysinfo c | |

src/h

| | | | | |
|---|---|---|---|---|
| Makefile | ctype h | gc h | pdef h | rt h |
| config gen | defs s | keyword h | pnames h | sysinfo h |
| config h | err h | memsize h | record h | |

src/icont

| | | |
|---|---|---|
| Makefile | icont c | ixhdr c |

src/iconx

| | | |
|---|---|---|
| Makefile | interp s | special s |
| init c | main c | start s |

src/lib

| | | | | |
|---|---|---|---|---|
| Makefile | coret s | esusp s | limit c | pfail s |
| bscan c | create c | field c | llist c | pret s |
| coact s | efail s | invoke s | lsusp s | psusp s |
| cofail s | escan c | keywd c | mkrec c | |

src/link

| | | | |
|---|---|---|---|
| Makefile | glob c | lcode c | lsym c |
| builtin c | ilink c | llex c | opcode c |
| datatype h | ilink h | lmem c | opcode h |

src/mc68000:

| | | | | |
|---|---|---|---|---|
| Makefile | csv.s | icont.c | ldfps.s | setbound.s |
| Setup* | defs.s | ilink.c | lsusp.s | special.s |
| arith.s | display.c | init.c | params.h | start.s |
| coact.s | efail.s | interp.s | pfail.s | suspend.s |
| cofail.s | esusp.s | invoke.s | pret.s | sweep.c |
| coret.s | fail.s | ixhdr.c | psusp.s | |
| create.c | gcollect.s | lcode.c | refresh.c | |

src/ops:

| | | | | |
|---|---|---|---|---|
| Makefile | lconcat.c | mult.c | numle.c | sect.c |
| asgn.c | lexeq.c | neg.c | numlt.c | size.c |
| bang.c | lexge.c | neqv.c | numne.c | subsc.c |
| cat.c | lexgt.c | nonnull.c | plus.c | swap.c |
| compl.c | lexle.c | null.c | power.c | tabmat.c |
| diff.c | lexlt.c | number.c | random.c | toby.c |
| div.c | lexne.c | numeq.c | rasgn.c | unions.c |
| eqv.c | minus.c | numge.c | refresh.c | value.c |
| inter.c | mod.c | numgt.c | rswap.c | |

src/pcix:

| | | | | |
|---|---|---|---|---|
| Makefile | csv.s | icont.c | ldfps.s | setbound.s |
| Setup | defs.s | ilink.c | lsusp.s | special.s |
| arith.s | display.c | init.c | params.h | start.s |
| coact.s | efail.s | interp.s | pfail.s | suspend.s |
| cofail.s | esusp.s | invoke.s | pret.s | sweep.c |
| coret.s | fail.s | ixhdr.c | psusp.s | |
| create.c | gcollect.s | lcode.c | refresh.c | |

src/pdp11:

| | | | | |
|---|---|---|---|---|
| Make.iconx | create.c | gcollect.s | lcode.c | refresh.c |
| Makefile | csv.s | icont.c | ldfps.s | setbound.s |
| Setup* | defs.s | ilink.c | lsusp.s | special.s |
| arith.s | display.c | init.c | params.h | start.s |
| coact.s | efail.s | interp.s | pfail.s | suspend.s |
| cofail.s | esusp.s | invoke.s | pret.s | sweep.c |
| coret.s | fail.s | ixhdr.c | psusp.s | |

src/pifncs:

| | | | |
|---|---|---|---|
| Makefile | iscope.c | seek.c | ttyctl.c |
| getenv.c | math.c | trig.c | |

src/proto:

| | | | | |
|---|---|---|---|---|
| Makefile | csv.s | icont.c | ldfps.s | setbound.s |
| Setup* | defs.s | ilink.c | lsusp.s | special.s |
| arith.s | display.c | init.c | params.h | start.s |
| coact.s | efail.s | interp.s | pfail.s | suspend.s |
| cofail.s | esusp.s | invoke.s | pret.s | sweep.c |
| coret.s | fail.s | ixhdr.c | psusp.s | |
| create.c | gcollect.s | lcode.c | refresh.c | |

src/ridge:

| | | | | |
|---|---|---|---|---|
| As* | create.c | gcollect.s | lcode.c | refresh.c |
| Makefile | csv.s | icont.c | ldfps.s | scomd |
| Setup* | defs.s | ilink.c | lsusp.s | setbound.s |
| arith.s | display.c | init.c | params.h | special.s |
| coact.s | efail.s | interp.s | pfail.s | start.s |
| cofail.s | esusp.s | invoke.s | pret.s | suspend.s |
| coret.s | fail.s | ixhdr.c | psusp.s | sweep.c |

src/rt:

| | | | | |
|---|---|---|---|---|
| Makefile | cvreal.c | dump.c | host.c | pow.c |
| addmem.c | cvstr.c | equiv.c | lexcmp.c | putstr.c |
| alc.c | dblocks.c | exp.c | locate.c | qtos.c |
| anycmp.c | defany.c | fail.s | log.c | setbound.s |
| arith.s | defcset.c | floor.c | memb.c | strprc.c |
| cplist.c | deffile.c | gc.c | memmon.c | suspend.s |
| ctype.c | defint.c | gcollect.s | mkint.c | sweep.c |
| cvcset.c | defshort.c | gcvt.c | mkreal.c | trace.c |
| cvint.c | defstr.c | gd.c | mksubs.c | tvkeys.c |
| cvnum.c | deref.c | getstr.c | numcmp.c | |
| cvpos.c | doasgn.c | hash.c | outimage.c | |

src/sys:

src/tran:

| | | | | |
|---|---|---|---|---|
| Makefile | icon.g | lex.h | optab | synerr.h |
| char.c | itran.c | lfile.h | optab.c | token.h |
| char.h | itran.h | lnklist.c | parse.c | tokens |
| code.c | keyword.c | mem.c | pscript | toktab.c |
| code.h | keywords | mkkeytab.icn | sym.c | tree.h |
| err.c | lex.c | mktoktab.icn | sym.h | |

src/unixpc:

| | | | | |
|---|---|---|---|---|
| Makefile | defs.s | init.c | moveq.s | setbound.s |
| Setup | display.c | interp.s | params.h | special.s |
| arith.s | efail.s | invoke.s | pfail.s | start.s |
| coact.s | esusp.s | ixhdr.c | pret.s | suspend.s |
| cofail.s | fail.s | lcode.c | pstart.s | sweep.c |
| coret.s | gcollect.s | ldfps.s | pstop.s | |
| create.c | icont.c | lsusp.s | psusp.s | |
| csv.s | ilink.c | moveq.c | refresh.c | |

src/vax:

| | | | | |
|---|---|---|---|---|
| Makefile | csv.s | icont.c | ldfps.s | setbound.s |
| Setup* | defs.s | ilink.c | lsusp.s | special.s |
| arith.s | display.c | init.c | params.h | start.s |
| coact.s | efail.s | interp.s | pfail.s | suspend.s |
| cofail.s | esusp.s | invoke.s | pret.s | sweep.c |
| coret.s | fail.s | ixhdr.c | psusp.s | |
| create.c | gcollect.s | lcode.c | refresh.c | |

test:

| | | | | |
|---|---|---|---|---|
| Makefile | org03.icn | std20.icn | std41.icn | std62.icn |
| README | org04.icn | std21.icn | std42.icn | std63.icn |
| Test* | std01.icn | std22.icn | std43.icn | std64.icn |
| buildt.icn | std02.icn | std23.icn | std44.icn | std65.icn |
| chk01.icn | std03.icn | std24.icn | std45.icn | std66.icn |
| chk02.icn | std04.icn | std25.icn | std46.icn | std67.icn |
| chk03.icn | std05.icn | std26.icn | std47.icn | std68.icn |
| dismem.icn | std06.icn | std27.icn | std48.icn | std69.icn |
| distr/ | std07.icn | std28.icn | std49.icn | std70.icn |
| ext01.icn | std08.icn | std29.icn | std50.icn | std71.icn |
| ext02.icn | std09.icn | std30.icn | std51.icn | std72.icn |
| ext03.icn | std10.icn | std31.icn | std52.icn | std73.icn |
| ext04.icn | std11.icn | std32.icn | std53.icn | std74.icn |
| ext05.icn | std12.icn | std33.icn | std54.icn | std75.icn |
| ext06.icn | std13.icn | std34.icn | std55.icn | std76.icn |
| ext07.icn | std14.icn | std35.icn | std56.icn | std77.icn |
| local/ | std15.icn | std36.icn | std57.icn | tlist |
| mem01.icn | std16.icn | std37.icn | std58.icn | |
| mem02.icn | std17.icn | std38.icn | std59.icn | |
| org01.icn | std18.icn | std39.icn | std60.icn | |
| org02.icn | std19.icn | std40.icn | std61.icn | |

test/distr:

| | | | | |
|---|---|---|---|---|
| chk01.out | std02.out | std22.out | std42.out | std62.out |
| chk02.out | std03.out | std23.out | std43.out | std63.out |
| chk03.out | std04.out | std24.out | std44.out | std64.out |
| exp01.out | std05.out | std25.out | std45.out | std65.out |
| exp02.out | std06.out | std26.out | std46.out | std66.out |
| exp03.out | std07.out | std27.out | std47.out | std67.out |
| ext01.out | std08.out | std28.out | std48.out | std68.out |
| ext02.out | std09.out | std29.out | std49.out | std69.out |
| ext03.out | std10.out | std30.out | std50.out | std70.out |
| ext04.out | std11.out | std31.out | std51.out | std71.out |
| ext05.out | std12.out | std32.out | std52.out | std72.out |
| ext06.out | std13.out | std33.out | std53.out | std73.out |
| ext07.out | std14.out | std34.out | std54.out | std74.out |
| mem01.out | std15.out | std35.out | std55.out | std75.out |
| mem02.out | std16.out | std36.out | std56.out | std76.out |
| org01.out | std17.out | std37.out | std57.out | std77.out |
| org02.out | std18.out | std38.out | std58.out | |
| org03.out | std19.out | std39.out | std59.out | |
| org04.out | std20.out | std40.out | std60.out | |
| std01.out | std21.out | std41.out | std61.out | |

test/local: