

GnuCOBOL Manual

for GnuCOBOL 3.1-dev

Keisuke Nishida, Roger While, Brian Tiffin, Simon Sobisch

Edition 3.1-dev

Updated for GnuCOBOL 3.1-dev

16 August 2019

GnuCOBOL (formerly OpenCOBOL) is a free COBOL compiler and runtime. `cobc` translates COBOL source to executable using intermediate C together with a designated C compiler and linker. `cobcrun` is a module loader to run generated modules, `libcob` provides the necessary runtime.

This manual corresponds to GnuCOBOL 3.1-dev.

Copyright © 2002-2012, 2014-2019 Free Software Foundation, Inc.

Written by Keisuke Nishida, Roger While, Brian Tiffin, Simon Sobisch.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Free Software Foundation.

Table of Contents

| | | |
|----------|-------------------------------------|-----------|
| 1 | Getting started | 1 |
| 1.1 | Hello, world! | 1 |
| 2 | Compile | 2 |
| 2.1 | Compiler options | 2 |
| 2.1.1 | Help options | 2 |
| 2.1.2 | Build target | 2 |
| 2.1.3 | Source format | 3 |
| 2.1.4 | Warning options | 3 |
| 2.1.5 | Configuration options | 4 |
| 2.1.6 | Listing options | 6 |
| 2.1.7 | Debug switches | 7 |
| 2.1.8 | Miscellaneous | 8 |
| 2.2 | Multiple sources | 8 |
| 2.2.1 | Static linking | 9 |
| 2.2.2 | Dynamic linking | 9 |
| 2.2.2.1 | Driver program | 9 |
| 2.2.2.2 | Compiling programs separately | 10 |
| 2.2.3 | Building library | 10 |
| 2.2.4 | Using library | 10 |
| 2.3 | C interface | 10 |
| 2.3.1 | Writing Main Program in C | 10 |
| 2.3.2 | Static linking with COBOL programs | 11 |
| 2.3.3 | Dynamic linking with COBOL programs | 12 |
| 2.3.4 | Static linking with C programs | 13 |
| 2.3.5 | Dynamic linking with C programs | 14 |
| 2.3.6 | Redirecting output to a (FILE *) | 14 |
| 3 | Customize | 15 |
| 3.1 | Customizing compiler | 15 |
| 3.2 | Customizing library | 15 |
| 4 | Optimize | 16 |
| 4.1 | Optimize options | 16 |
| 4.2 | Optimize call | 16 |
| 4.3 | Optimize binary | 16 |
| 5 | Debug | 17 |
| 5.1 | Debug options | 17 |
| 6 | Non-standard extensions | 18 |
| 6.1 | SELECT ASSIGN TO | 18 |
| 6.1.1 | Literal file | 18 |
| 6.1.2 | <variable> | 18 |
| 6.1.3 | <environment variable> | 18 |
| 6.2 | Indexed file packages | 18 |

| | | |
|-------------------|------------------------------------------------|-----------|
| 6.3 | Extended ACCEPT statement | 18 |
| 6.3.1 | LINE | 19 |
| 6.3.2 | COLUMN | 19 |
| 6.3.3 | AUTO-SKIP | 19 |
| 6.3.4 | BACKGROUND-COLOR | 19 |
| 6.3.5 | BELL | 19 |
| 6.3.6 | BLINK | 19 |
| 6.3.7 | FOREGROUND-COLOR | 19 |
| 6.3.8 | LOWLIGHT | 19 |
| 6.3.9 | PROMPT | 19 |
| 6.3.10 | PROTECTED | 20 |
| 6.3.11 | SIZE | 20 |
| 6.3.12 | UPDATE | 20 |
| 6.3.13 | ON EXCEPTION | 20 |
| 6.3.14 | NOT ON EXCEPTION | 20 |
| 6.4 | ACCEPT special keys | 20 |
| 6.4.1 | Arrow keys | 20 |
| 6.4.2 | Backspace key | 20 |
| 6.4.3 | Delete keys | 21 |
| 6.4.4 | End key | 21 |
| 6.4.5 | Home key | 21 |
| 6.4.6 | Insert key | 21 |
| 6.4.7 | Tab keys | 21 |
| 6.5 | Extended DISPLAY statement | 21 |
| 6.5.1 | BELL | 21 |
| 6.5.2 | BLANK | 21 |
| 6.5.3 | ERASE | 22 |
| 6.5.4 | SIZE | 22 |
| 6.5.5 | Figurative Constants | 22 |
| 6.6 | CONTENT-LENGTH | 22 |
| 6.7 | CONTENT-OF | 23 |
| 7 | System Routines | 24 |
| 7.1 | CBL_GC_GETOPT | 24 |
| 7.2 | CBL_GC_HOSTED | 25 |
| 7.3 | CBL_GC_NANOSLEEP | 28 |
| 7.4 | CBL_GC_FORK | 28 |
| 7.5 | CBL_GC_WAITPID | 29 |
| Appendix A | Compiler cobc options | 30 |
| A.1 | Options | 30 |
| A.2 | Warning options | 32 |
| A.3 | Compiler options | 33 |
| A.4 | Compiler dialect configuration options | 33 |
| Appendix B | Reserved Words | 39 |
| B.1 | Common reserved words | 39 |
| B.2 | Extra (obsolete) context sensitive words | 56 |
| B.3 | Internal registers | 57 |
| Appendix C | Intrinsic Functions | 58 |

| | | |
|-------------------|---------------------------------------|-----------|
| Appendix D | System routines | 61 |
| Appendix E | System names | 63 |
| E.1 | System names: device | 63 |
| E.2 | System names: feature | 63 |
| E.3 | System names: switch | 63 |
| Appendix F | Compiler Configuration | 64 |
| Appendix G | Module loader cobcrun options | 69 |
| Appendix H | Runtime configuration | 70 |
| H.1 | General instructions | 70 |
| H.2 | General environment | 70 |
| H.3 | Call environment | 72 |
| H.4 | File I/O | 73 |
| H.5 | Screen I/O | 74 |
| H.6 | Report I/O | 77 |
| Appendix I | GNU Free Documentation License | 78 |

1 Getting started

1.1 Hello, world!

This is a sample program that displays “Hello, world!”:

```
----- hello.cob -----
      * Sample COBOL program
      IDENTIFICATION DIVISION.
      PROGRAM-ID. hello.
      PROCEDURE DIVISION.
      DISPLAY "Hello, world!".
      STOP RUN.
-----
```

The compiler, `cobc`, is executed as follows:

```
$ cobc -x hello.cob
$ ./hello
Hello, world!
```

The executable file name (`hello` in this case) is determined by removing the extension from the source file name.

You can specify the executable file name by specifying the compiler option `-o` as follows:

```
$ cobc -x -o hello-world hello.cob
$ ./hello-world
Hello, world!
```

The program can be written in a more modern style, with free format code, inline comments, the `GOBACK` verb and an optional `END-DISPLAY` terminator:

```
----- hellonew.cob -----
*> Sample GnuCOBOL program
identification division.
program-id. hellonew.
procedure division.
display
    "Hello, new world!"
end-display
goback.
-----
```

To compile free-format code, you must use the compiler option `-free`.

```
$ cobc -x -free hellonew.cob
$ ./hellonew
Hello, new world!
```

2 Compile

This chapter describes how to compile COBOL programs using GnuCOBOL.

2.1 Compiler options

The compiler `cobc` accepts the options described in this section. The compiler arguments follow the general syntax `cobc options file [file ...]`. A complete list of options can be displayed by using the option `--help`.

2.1.1 Help options

The following switches display information about the compiler:

- `--help, -h`
Display help screen (see Appendix A [Appendix A], page 30). No further actions will be taken.
- `--version`
Display compiler version, author package date and executable build date. `-V` will also display version. No further actions will be taken.
- `--info`
Display build information along with the default and current compiler configurations. No further actions will be taken except for further display options.
- `-v`
Verbosely display the programs invoked during compilation.
- `--list-reserved`
Display reserved words (see Appendix B [Appendix B], page 39). A Yes/No output shows if the word is supported¹, context sensitive and its aliases. The given options for reserved words specified for example by option `-std=dialect` will be taken into account. No further actions will be taken except for further display options.
- `--list-intrinsics`
Display intrinsic functions (see Appendix C [Appendix C], page 58). A Y/N field shows if the function is implemented. No further actions will be taken except for further display options.
- `--list-system`
Display system routines (see Appendix D [Appendix D], page 61). No further actions will be taken except for further display options.
- `--list-mnemonics`
Display mnemonic names (see Appendix E [Appendix E], page 63). No further actions will be taken except for further display options.

2.1.2 Build target

The compiler `cobc` treats files like `*.cob`, `*.cbl` as COBOL source code, `*.c` as C source code, `*.o` as object code, `*.i` as preprocessed code and `*.so` as dynamic modules and knows how to handle such files in the generation, compilation, and linking steps.

The special input name `-` takes input from `stdin` which is assumed to be COBOL source, and uses a default output name of `a.out` (or `a.so/c/o/i`, selected as appropriate) for the build type.

By default, the compiler builds a dynamically loadable module.

¹ Support may be partial or complete.

The following options specify the target type produced by the compiler:

- E Preprocess only: compiler directives are executed, comment lines are removed and COPY statements are expanded. The output is saved in file **.i*.
- C Translation only. COBOL source files are translated into C files. The output is saved in file **.c*.
- S Compile only. Translated C files are compiled by the C compiler to assembler code. The output is saved in file **.s*.
- c Compile and assemble. This is equivalent to `cc -c`. The output is saved in file **.o*.
- m Compile, assemble, and build a dynamically loadable module (i.e., a shared library). The output is saved in file **.so*.² This is the default behaviour.
- b Compile, assemble, and combine all input files into a single dynamically loadable module. Unless `-o` is also used, the output is saved using the first filename as **.so*.
- x Include the main function in the output, creating an executable image. The main entry point being the first program in the file.
This option takes effect at the translation stage. If you give this option with `-C`, you will see the main function at the end of the generated C file.
- j, -job, -j=args, -job=args
Run job after compilation. Either from executable with `-x`, or with `cobcrun` when compiling a module. Optional arguments *args*, if given, are passed to the program or module command line.
- I *directory*
Add *directory* to copy/include search path.
- L *directory*
Add *directory* to library search path.
- l *lib* Link the library *lib*.
- D *define* Pass *define* to the COBOL compiler.
- o *file* Place the output into *file*.

2.1.3 Source format

GnuCOBOL supports both fixed and free source format. The default format is the fixed format. This can be overridden either by the `>>SOURCE [FORMAT] [IS] {FIXED|FREE}` directive, or by one of the following options:

- free, -F Free format. The program-text area starts in column 1 and continues till the end of line (effectively 255 characters in GnuCOBOL).
- fixed Fixed format. Source code is divided into: columns 1-6, the sequence number area; column 7, the indicator area; columns 8-72, the program-text area; and columns 72-80 as the reference area.³

2.1.4 Warning options

- W Enable every possible warning. This includes more information than `-Wall` would normally provide.

² The extension varies depending on your host.

³ Historically, fixed format was based on 80-character punch cards.

- `-Wall` Enable all common warnings.
- `-Wwarning`
 Enable single warning *warning*.
- `-Wno-warning`
 Disable single warning *warning*.
- `-Warchaic`
 Warn if archaic features are used, such as continuation lines or the `NEXT SENTENCE` statement.
- `-Wcall-params`
 Warn if non-01/77-level items are used as arguments in a `CALL` statement. This is *not* set with `-Wall`.
- `-Wcolumn-overflow`
 Warn if text after column 72 in `FIXED` format. This is *not* set with `-Wall`.
- `-Wconstant`
 Warn inconsistent constant
- `-Wimplicit-define`
 Warn if implicitly defined data items are used.
- `-Wlinkage`
 Warn dangling `LINKAGE` items. This is *not* set with `-Wall`.
- `-Wobsolete`
 Warn if obsolete features are used.
- `-Wparentheses`
 Warn about any lack of parentheses around `AND` within `OR`.
- `-Wredefinition`
 Warn about incompatible redefinitions of data items.
- `-Wstrict-typing`
 Warn about type mismatch strictly.
- `-Wterminator`
 Warn about the lack of scope terminator `END-XXX`. This is *not* set with `-Wall`.
- `-Wtruncate`
 Warn on possible field truncation. This is *not* set with `-Wall`.
- `-Wunreachable`
 Warn if statements are unreachable. This is *not* set with `-Wall`.

2.1.5 Configuration options

- `-std=diagnostic`
 Compiler uses the given *dialect* to determine certain compiler features and warnings. See Appendix F [Compiler Configuration], page 64, and `config/*.conf`.
 Note: The GnuCOBOL compiler tries to limit both the feature-set and reserved words to the specified compiler when the "strict" dialects are used. COBOL sources compiled with these dialects are therefore likely to compile with the specified compiler and vice versa: sources that were compiled on the specified compiler should compile without any issues with GnuCOBOL.
 With the "non-strict" dialects GnuCOBOL will activate the complete feature-set where it doesn't directly conflict with the specified dialect, including reserved words.

COBOL sources compiled with these dialects therefore may work only with GnuCOBOL. COBOL sources may need a change because of reserved words in GnuCOBOL, otherwise offending words *word-1* and *word-2* may be removed by `-fno-reserved=word-1,word-1`.

COBOL-85, X/Open COBOL, COBOL 2002 and COBOL 2014 are always "strict".

`-std=default`

GnuCOBOL dialect, supporting many of the COBOL 2002 and COBOL 2014 features, many extensions found in other dialects and its own feature-set

`-std=cobol85`

COBOL-85 without any extensions other than the amendment Intrinsic Function Module (1989), source compiled with this dialect is likely to compile with most COBOL compilers

`-std=xopen`

X/Open COBOL (based on COBOL-85) without any vendor extensions, source compiled with this dialect is likely to compile with most COBOL compilers; will warn items that "should not be used in a conforming X/Open COBOL source program"

`-std=cobol2002, -std=cobol2014`

COBOL 2002 / COBOL 2014 without any vendor extensions, use `-Warchaic` and `-Wobsolete` if archaic/obsolete features should be flagged

`-std=ibm-strict, -std=ibm`

IBM compatible

`-std=mvs-strict, -std=mvs`

MVS compatible

`-std=mf-strict, -std=mf`

Micro Focus compatible

`-std=bs2000-strict, -std=bs2000`

BS2000 compatible

`-std=acu-strict, -std=acu`

ACUCOBOL-GT compatible

`-std=rm-strict, -std=rm`

RM/COBOL compatible

`-conf=<file>`

User-defined dialect configuration. See `-std=dialect` above.

You can override each single configuration entry by using compiler configuration options on the command line.

Examples:

`-frelax-syntax-checks`

`-frenames-uncommon-levels=warning`

`-fnot-reserved=CHAIN,SCREEN`

`-ftab-width=4`

See Appendix A [Compiler `cobc` options], page 30.

2.1.6 Listing options

-t=file Generate and place the standard print listing into *file*.

-T=file Generate and place a wide print listing into **file*.

```
--tlines=lines
```

Specify lines per page in print listing, default = 55. Set to zero for no additional page breaks.

-ftsymbols

Generate symbol table in listing.

`-fno-theader`

Suppress all headers from listing while keeping page breaks.

`-fno-tmessages`

Suppress warning and error summary from listing.

-fno-tssource

Suppress actual source from listing (for example to only produce the cross-reference).

`-P, -Pdirectory, -P=file`

Generate and place a preprocessed listing (old format) into *filename.lst*, *directory/filename.lst*, *file*.

-Xref

-X Generate cross reference in the listing.

Here is an example program listing with the options `-t -ftsymbols`:

GnuCOBOL 3.0.0 test.cbl

Mon May 14 10:23:45 2018 Page 0001

| LINE | PG/LN | A...B..... |
|------|-------|------------|
|------|-------|------------|

```

000001      IDENTIFICATION      DIVISION.
000002      PROGRAM-ID.          prog.
000003      ENVIRONMENT DIVISION.
000004      CONFIGURATION SECTION.
000005      DATA                  DIVISION.
000006      WORKING-STORAGE SECTION.
000007      COPY 'values.cpy'.
000001C      78 I      VALUE 20.
000002C      78 J      VALUE 5000.
000003C      78 M      VALUE 5.
000008      01 SETUP-REC.
000009           05 FL1          PIC X(04).
000010           05 FL2          PIC ZZZZZ.
000011           05 FL3          PIC 9(04).
000012           05 FL4          PIC 9(08) COMP.
000013           05 FL5          PIC 9(04) COMP-4.
000014           05 FL6          PIC Z,ZZZ.99.
000015           05 FL7          PIC S9(05) SIGN LEADING SEPARATE.
000016           05 FL8          PIC X(04).
000017           05 FL9 REDEFINES FL8 PIC 9(04).
000018           05 FLA.
000019           10 FLB OCCURS I TIMES.
000020           15 FLC PIC X(02).

```

```

000021          10  FLD  PIC X(20).
000022          05  FLD1  PIC X(100).
000023          05  FLD2 OCCURS M TO J TIMES DEPENDING ON FL5.
000024          10  FILLER PIC X(01).
000025          05  FLD3  PIC X(3).
000026          05  FLD4  PIC X(4).
000027  PROCEDURE      DIVISION.
000028          STOP RUN.

```

The first part of the listing lists the program text. If the program text is a COPY the line number reflects the COPY line number and is appended with a 'C'.

When the wide list option -T is specified, the SEQUENCE columns (for fixed-form reference-format) are included in the listing.

The second part of the listing file is the listing of the Symbol Table:

```

GnuCOBOL 3.0.0   test.cbl                               Mon May 14 10:23:45 2018   Page 0002

```

| SIZE | TYPE | LVL | NAME | PICTURE |
|------|----------------|-----|-----------|------------------|
| 5204 | GROUP | 01 | SETUP-REC | |
| 0004 | ALPHANUMERIC | 05 | FL1 | X(04) |
| 0005 | ALPHANUMERIC | 05 | FL2 | ZZZZZ |
| 0004 | ALPHANUMERIC | 05 | FL3 | 9(04) |
| 0004 | NUMERIC | 05 | FL4 | 9(08) COMP |
| 0002 | NUMERIC | 05 | FL5 | 9(04) COMP |
| 0008 | ALPHANUMERIC | 05 | FL6 | Z,ZZZ.99 |
| 0006 | ALPHANUMERIC | 05 | FL7 | S9(05) |
| 0004 | ALPHANUMERIC | 05 | FL8 | X(04) |
| 0004 | ALPHANUMERIC-R | 05 | FL9 | 9(04) |
| 0060 | ALPHANUMERIC | 05 | FLA | |
| 0040 | ALPHANUMERIC | 10 | FLB | OCCURS 20 |
| 0002 | ALPHANUMERIC | 15 | FLC | X(02) |
| 0020 | ALPHANUMERIC | 10 | FLD | X(20) |
| 0100 | ALPHANUMERIC | 05 | FLD1 | X(100) |
| 5000 | ALPHANUMERIC | 05 | FLD2 | OCCURS 5 TO 5000 |
| 0001 | ALPHANUMERIC | 10 | FILLER | X(01) |
| 0003 | ALPHANUMERIC | 05 | FLD3 | X(3) |
| 0004 | ALPHANUMERIC | 05 | FLD4 | X(4) |

If the symbol redefines another variable the TYPE is marked with 'R'. If the symbol is an array the OCCURS phrase is in the PICTURE field.

The last part of the listing file is the summary of warnings and error in the compilation group:

```

0 warnings in compilation group
2 errors in compilation group

```

2.1.7 Debug switches

- debug, -d Enable all run-time error checks.
- g Produce C debugging information in the output.
- ftrace Generate trace code (log executed procedures, if tracing is enabled).
- ftraceall Generate trace code (log executed procedures and statements, if tracing is enabled).

- `-fsource-location`
Generate source location code (implied by `-debug` or `-g`).
- `-fstack-check`
Enable PERFORM stack checking (implied by `-debug` or `-g`).
- `-fdebugging-line`
Enable debugging lines (D in indicator column; >>D directive).
- `-O`
Enable optimization of code size and execution speed. See your C compiler documentation, for example `man gcc` for details.
- `-O2`
Optimize even more.
- `-Os`
Optimize for size. Optimizer will favour code size over execution speed.
- `-fnotrunc`
Do not truncate binary fields according to PICTURE.

2.1.8 Miscellaneous

- `-ext <extension>`
Add default file extension.
- `-fsyntax-only`
Check syntax only; don't emit any output.
- `-fmfcomment`
Treat lines with * or / in column 1 as comment (fixed-form reference-format only).
- `-acucomment`
Treat | as an inline comment marker.
- `-fsign=ASCII`
Numeric display sign ASCII (default on ASCII machines).
- `-fsign=EBCDIC`
Numeric display sign EBCDIC (default on EBCDIC machines).
- `-fintrinsics=[ALL|intrinsic function name(,name,...)]`
Allow use of all or specific intrinsic functions without FUNCTION keyword.
Note: defining this within your source with CONFIGURATION SECTION. REPOSITORY. is preferred.
- `-ffold-copy=LOWER`
Fold COPY subject to lower case (default no transformation).
- `-ffold-copy=UPPER`
Fold COPY subject to upper case (default no transformation).
- `-save-temps(=<dir>)`
Save intermediate files (by default, in current directory).
- `-fimplicit-init`
Do automatic initialization of the COBOL runtime system.

2.2 Multiple sources

This section describes how to compile a program from multiple source files.

This section also describes how to build a shared library that can be used by any COBOL program and how to use external libraries in COBOL programs.

2.2.1 Static linking

The easiest way of combining multiple files is to compile them into a single executable.

One way is to compile all the files in one command:

```
$ cobc -x -o prog main.cob subr1.cob subr2.cob
```

Another way is to compile each file with the option `-c`, and link them at the end. The top-level program must be compiled with the option `-x`.

```
$ cobc -c subr1.cob
$ cobc -c subr2.cob
$ cobc -c -x main.cob
$ cobc -x -o prog main.o subr1.o subr2.o
```

You can link C routines as well using either method:

```
$ cobc -o prog main.cob subrs.c
```

or

```
$ cobc -c subrs.c
$ cobc -c -x main.cob
$ cobc -x -o prog main.o subrs.o
```

Any number of functions can be contained in a single C file.

The linked programs will be called dynamically; that is, the symbol will be resolved at run time. For example, the following COBOL statement

```
CALL "subr" USING X.
```

will be converted into equivalent C code like this:

```
int (*func)() = cob_resolve("subr");
if (func != NULL)
    func (X);
```

With the compiler option `-fstatic-call`, more efficient code will be generated:

```
subr(X);
```

Please notice that this option only takes effect when the called program name is in a literal (like `CALL "subr"`). With a data name (like `CALL SUBR`), the program is still called dynamically.

2.2.2 Dynamic linking

There are two methods to achieve this: a driver program, or compiling the main program and subprograms separately.

2.2.2.1 Driver program

Compile all programs with the option `-m`:

```
$ cobc -m main.cob subr.cob
```

This creates the shared object files `main.so` and `subr.so`.⁴

Before running the main program, install the module files in your library directory:

```
$ cp subr.so /your/cobol/lib
```

Set the runtime variable `COB_LIBRARY_PATH` to your library directory, and run the main program:

```
$ export COB_LIBRARY_PATH=/your/cobol/lib
```

(Please notice: You may set the variable via a runtime configuration file, see Appendix H [Runtime Configuration], page 70. You may also set the variable to directly point to the directory where you compiled the sources.)

⁴ The extension used depends on your operating system.

Now execute your program:

```
$ cobcrun main
```

2.2.2.2 Compiling programs separately

The main program is compiled as usual:

```
$ cobc -x -o main main.cob
```

Subprograms are compiled with the option `-m`:

```
$ cobc -m subr.cob
```

This creates a module file `subr.so`⁵.

Before running the main program, install the module files in your library directory:

```
$ cp subr.so /your/cobol/lib
```

Now, set the environment variable `COB_LIBRARY_PATH` to your library directory, and run the main program:

```
$ export COB_LIBRARY_PATH=/your/cobol/lib
$ ./main
```

2.2.3 Building library

You can build a shared library by combining multiple COBOL programs and even C routines:

```
$ cobc -c subr1.cob
$ cobc -c subr2.cob
$ cc -c subr3.c
$ cc -shared -o libsubrs.so subr1.o subr2.o subr3.o
```

2.2.4 Using library

You can use a shared library by linking it with your main program.

Before linking the library, install it in your system library directory:

```
$ cp libsubrs.so /usr/lib
```

or install it somewhere else and set `LD_LIBRARY_PATH`:

```
$ cp libsubrs.so /your/cobol/lib
$ export LD_LIBRARY_PATH=/your/cobol/lib
```

Then, compile the main program, linking the library as follows:

```
$ cobc -x main.cob -L/your/cobol/lib -lsubrs
```

2.3 C interface

This chapter describes how to combine C programs with COBOL programs.

2.3.1 Writing Main Program in C

Include `libcob.h` in your C program and call `cob_init` before using any COBOL module. Do a cleanup afterwards, either by calling `cob_stop_run` (if your program should terminate) or by calling `cob_tidy` (if your program should execute further on without any more COBOL calls, calling both functions in this sequence can be done multiple times).

```
#include <libcob.h>

int
main (int argc, char **argv)
```

⁵ The extension used depends on your operating system.

```

{
    /* initialize your program */
    ...

    /* initialize the COBOL run-time library */
    cob_init (argc, argv);

    /* rest of your program */
    ...

    /* Clean up and terminate - This does not return */
    cob_stop_run (return_status);
}

```

You can write `cobc_init(0, NULL)`; if you do not want to pass command line arguments to COBOL.

You can compile your C program as follows:

```
cc -c `cob-config --cflags` main.c
```

The compiled object must be linked with `libcob` as follows:

```
cc -o main main.o `cob-config --libs`
```

2.3.2 Static linking with COBOL programs

Let's call the following COBOL module from a C program:

```

----- say.cob -----
    IDENTIFICATION DIVISION.
    PROGRAM-ID. say.
    ENVIRONMENT DIVISION.
    DATA DIVISION.
    LINKAGE SECTION.
    01 hello PIC X(7).
    01 world PIC X(6).
    PROCEDURE DIVISION USING hello world.
        DISPLAY hello world.
        EXIT PROGRAM.
-----

```

This program accepts two arguments, displays them, and exits.

From the viewpoint of C, this is equivalent to a function having the following prototype:

```
extern int say(char *hello, char *world);
```

So, your main program will look like as follows:

```

----- hello.c -----
#include <libcob.h>

extern int say(char *hello, char *world);

int
main()
{
    int ret;
    char hello[8] = "Hello, ";
    char world[7] = "world!";

```



```

/* initialize the COBOL run-time library */
cob_init(0, NULL);

/* call the static module and store its return code */
ret = say(hello, world);

/* shutdown the COBOL run-time library, keep program running */
(void)cob_tidy();

return ret;
}
-----

```

Compile these programs as follows:

```

$ cc -c 'cob-config --cflags' hello.c
$ cobc -c -static say.cob
$ cobc -x -o hello hello.o say.o
$ ./hello
Hello, world!

```

2.3.3 Dynamic linking with COBOL programs

You can find a COBOL module having a specific name by using the C function `cob_resolve`, which takes the module name as a string and returns a pointer to the module function.

`cob_resolve` returns `NULL` if there is no module. In this case, the function `cob_resolve_error` returns the error message.

Let's see an example:

```

---- hello-dynamic.c -----
#include <libcob.h>

static int (*say)(char *hello, char *world);

int main()
{
    int ret;
    char hello[8] = "Hello, ";
    char world[7] = "world!";

    /* initialize the COBOL run-time library */
    cob_init(0, NULL);

    /* Find the module with PROGRAM-ID "say". */
    say = cob_resolve("say");

    /* If there is no such module, show error and exit. */
    if(say == NULL) {
        fprintf(stderr, "%s\n", cob_resolve_error());
        exit(1);
    }

    /* Call the module found ... */
    ret = say(hello, world);
}

```

```

    /* ...and exit with the return code. */
    cob_stop_run(ret);
}
-----

```

Compile these programs as follows:

```

$ cc -c 'cob-config --cflags' hello-dynamic.c
$ cobc -x -o hello hello-dynamic.o
$ cobc -m say.cob
$ export COB_LIBRARY_PATH=.
$ ./hello
Hello, world!

```

2.3.4 Static linking with C programs

Let's call the following C function from COBOL:

```

---- say.c -----
int say(char *hello, char *world)
{
    int i;
    for(i = 0; i < 7; i++)
        putchar(hello[i]);
    for(i = 0; i < 6; i++)
        putchar(world[i]);
    putchar('\n');
    return 0;
}
-----

```

This program is equivalent to the program in `say.cob` above.

Note that, unlike C, the arguments passed from COBOL programs are not terminated by the null character (i.e., `'\0'`).

You can call this function in the same way you call COBOL programs:

```

---- hello.cob -----
IDENTIFICATION DIVISION.
PROGRAM-ID. hello.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 hello PIC X(7) VALUE "Hello, ".
01 world PIC X(6) VALUE "world!".
PROCEDURE DIVISION.
CALL "say" USING hello world.
STOP RUN.
-----

```

Compile these programs as follows:

```

$ cc -c say.c
$ cobc -c -static -x hello.cob
$ cobc -x -o hello hello.o say.o
$ ./hello
Hello, world!

```

2.3.5 Dynamic linking with C programs

You can create a dynamically-linked module from a C program by passing an option `-shared` to the C compiler:

```
$ cc -shared -o say.so say.c
$ cobc -x hello.cob
$ export COB_LIBRARY_PATH=.
$ ./hello
Hello, world!
```

2.3.6 Redirecting output to a (FILE *)

From a module written in C you can call `cob_set_runtime_option` to set the exact (FILE *) which is used to write trace data to. In `common.h` is the following:

```
enum cob_runtime_option_switch {
    COB_SET_RUNTIME_TRACE_FILE           /* 'p' is FILE * */
    COB_SET_RUNTIME_DISPLAY_PRINTER_FILE /* 'p' is FILE * */
    COB_SET_RUNTIME_RESCAN_ENV           /* rescan environment variables */
    COB_SET_RUNTIME_DISPLAY_PUNCH_FILE   /* 'p' is FILE * */
};
COB_EXPIMP void cob_set_runtime_option (enum cob_runtime_option_switch opt, void *)
```

So from you C code you can tell the GnuCOBOL runtime to redirect TRACE output by:

```
cob_set_runtime_option (COB_SET_RUNTIME_TRACE_FILE, (void*)((FILE*)myfd));
```

You could also redirect all DISPLAY UPON PRINTER output to a file by:

```
cob_set_runtime_option (COB_SET_RUNTIME_DISPLAY_PRINTER_FILE, (void*)((FILE*)myfd));
```

You could also redirect all DISPLAY UPON SYSPUNCH output to a file by:

```
cob_set_runtime_option (COB_SET_RUNTIME_DISPLAY_PUNCH_FILE, (void*)((FILE*)myfd));
```

Another routine can be used to return the current value of the option.

```
COB_EXPIMP void *cob_get_runtime_option (enum cob_runtime_option_switch opt);
```

3 Customize

3.1 Customizing compiler

These settings are effective at compile-time.

Environment variables (default value in brackets):

COB_CC C compiler ("gcc")

COB_CFLAGS
Flags passed to the C compiler ("-I\$(PREFIX)/include")

COB_LDFLAGS
Flags passed to the C compiler ("")

COB_LIBS Standard libraries linked with the program ("-L\$(PREFIX)/lib -lcob")

COB_LDADD
Additional libraries linked with the program ("")

3.2 Customizing library

These settings are effective at run-time. You can set them either via the environment or by a runtime configuration file.

To set the global runtime configuration file export **COB_RUNTIME_CONFIG** to point to your configuration file. To set an explicit runtime configuration file for a single run via **cobcrun** you can use its option **-c file**, **--config=file**.

For displaying the current runtime settings you can use the option **-r**, **--runtime-env** of **cobcrun**.

For a complete list of runtime variables, aliases, their default values and options to set them see Appendix H [Runtime Configuration], page 70.

4 Optimize

4.1 Optimize options

There are five compiler options for optimization: `-O0`, `-O`, `-Os`, `-O2`, `-O3`. These options enable optimization at both translation (from COBOL to C) and compilation (C to assembly) levels.

Currently, there is no difference between these optimization options at the translation level.

The option `-O`, `-Os` or `-O2` is passed to the C compiler as is and used for C level optimization.

4.2 Optimize call

When a `CALL` statement is executed, the called program is linked at run time. By specifying the compiler option `-fstatic-call`, you can statically link the program at compile time and call it efficiently. (see Section 2.2.1 [Static linking], page 9)

4.3 Optimize binary

By default, data items of usage `binary` or `comp` are stored in big-endian form. On those machines whose native byte order is little-endian, this is not quite efficient.

If you prefer, you can store binary items in the native form of your machine. Set the config option `binary-byteorder` to `native` in your config file (see Chapter 3 [Customize], page 15).

In addition, setting the option `binary-size` to `2-4-8` or `1-2-4-8` is more efficient than others.

5 Debug

5.1 Debug options

The compiler option `-debug` can be used during the development of your programs. It enables all run-time error checking, such as subscript boundary checks and numeric data checks, and displays run-time errors with source locations.

6 Non-standard extensions

6.1 SELECT ASSIGN TO

A file may be assigned to a literal file, a file in a variable, or a file in an environment variable.

6.1.1 Literal file.

Assign to a literal file.

Select *file* assign to `"/tmp/myfile.txt"`.

6.1.2 <variable>

Assign to a file which name is read from a variable.

Select *file* assign to `my-file`.

```
01 my-file          pic x(512).
```

Move `"/tmp/myfile.txt"` to `my-file`.

Open output `<file>`.

6.1.3 <environment variable>

Assign to a file in an environment variable.

```
export myfile=/tmp/myfile.txt
```

Select *file* assign to external `myfile`.

6.2 Indexed file packages

<This section is in progress.>

6.3 Extended ACCEPT statement

Extended ACCEPT statements allow for full control of items accepted from the screen. Items accept by line and column positioning.

All commands following WITH are optional.

```
ACCEPT variable-1
```

```
  LINE variable-2 | literal-1 COLUMN variable-3 | literal-2
```

```
  WITH
```

```
    AUTO-SKIP | AUTO
```

```
    BACKGROUND-COLOR variable-4 | literal-3
```

```
    BELL | BEEP
```

```
    BLINK
```

```
    FOREGROUND-COLOR variable-5 | literal-4
```

```
    LOWLIGHT | HIGHLIGHT
```

```
    PROMPT
```

```
    PROTECTED
```

```
    SIZE [IS] variable-6 | literal-5
```

```
    UPDATE
```

```
  ON EXCEPTION
```

```
    exception processing
```

```

NOT ON EXCEPTION
    normal processing
END-ACCEPT.

```

6.3.1 LINE

The line number of *variable-2* or *literal-1* to accept the field.

6.3.2 COLUMN

The column number of *variable-3* or *literal-2* to accept the field.

6.3.3 AUTO-SKIP

The word `AUTO` may be used for `AUTO-SKIP`.

With this option the `ACCEPT` statement returns after the last character is typed at the end of the field. This is the same as if the Enter key were pressed.

Without this option the cursor remains at the end of the field and waits for the user to press Enter.

The Right-Arrow key returns from the end of the field. The Left-Arrow key returns from the beginning. See Section 6.4 [ACCEPT special], page 20.

The Alt-Right-Arrow and Alt-Left-Arrow keys never `AUTO-SKIP`.

6.3.4 BACKGROUND-COLOR

The background color is the color used behind the characters.

Variable-4 or *literal-3* must be numeric. See file `screenio.cpy` for the color assignments to *variable-4* or *literal-3*.

6.3.5 BELL

The word `BEEP` may be used for `BELL`.

The system beeps when the cursor moves to accept from this field. On some systems, there is no sound. Some other method may indicate a beep, such a flashing screen or pop up window.

6.3.6 BLINK

The field blinks while the user enters the data. This can help small menu selection fields to stand out.

6.3.7 FOREGROUND-COLOR

The foreground color is the color used for the characters.

Variable-5 or *literal-4* must be numeric. See file `screenio.cpy` for the color assignments to *variable-5* or *literal-4*.

6.3.8 LOWLIGHT

The `LOWLIGHT` and `HIGHLIGHT` phrases vary the intensity of the field.

`LOWLIGHT` displays with lower intensity and `HIGHLIGHT` displays with higher intensity. Having neither `LOWLIGHT` nor `HIGHLIGHT` displays at normal intensity.

These may have different levels of intensity, if at all, depending on the make and model of the screens.

6.3.9 PROMPT

Display the field with prompt characters as the cursor moves to accept from this field.

6.3.10 PROTECTED

PROTECTED is ignored.

6.3.11 SIZE

The size of *variable-1* to accept from the screen.

Variable-6 or *literal-5* must be numeric.

SIZE <greater than zero>

If *variable-6* or *literal-5* is less than the length of *variable-1* then only the SIZE number of characters accept into the field. *Variable-1* pads with spaces after SIZE to the end of the field.

If *variable-6* or *literal-5* is greater than *variable-1*, then the screen pads with spaces after *variable-1* to the SIZE length.

SIZE ZERO

<SIZE option not specified>

The *variable-1* accepts to its field length.

6.3.12 UPDATE

The contents of *variable-1* displays on the screen as the ACCEPT begins. This allows the user to update the field without having to type it all again.

Without this option, the ACCEPT field is always blank.

6.3.13 ON EXCEPTION

Check the special register `cob-crt-status` for the special key that was pressed. This includes Escape, Tab, Back-Tab, F-keys, arrows, etc... See `screenio.cpy` for the values.

6.3.14 NOT ON EXCEPTION

Reset any F-key indicator because no special key was pressed.

6.4 ACCEPT special keys

Special keys are available for extended ACCEPT statements.

The `COB-CRT-STATUS` values are in the `screenio.cpy` copy file.

6.4.1 Arrow keys

The Left-Arrow key moves the cursor to the left. Without `AUTO-SKIP` the cursor stops at the beginning of the field. With `AUTO-SKIP` it returns with the `COB-SCR-KEY-LEFT` value of 2009. See Section 6.3 [Extended ACCEPT], page 18.

The Alt-Left-Arrow key is the same as Left-Arrow except that it never returns, even for `AUTO-SKIP`.

The Right-Arrow key moves the cursor to the right. Without `AUTO-SKIP` the cursor stops at the end of the field. With `AUTO-SKIP` it returns with the `COB-SCR-KEY-RIGHT` value of 2010. See Section 6.3 [Extended ACCEPT], page 18.

The Alt-Right-Arrow key is the same as Right-Arrow except that it never returns, even for `AUTO-SKIP`.

6.4.2 Backspace key

The Backspace key moves the cursor, and the remainder of the text, to the left.

6.4.3 Delete keys

The Delete key deletes the cursor's character and moves the remainder of the text to the left. The cursor does not move.

The Alt-Delete key deletes all text from the cursor to the end of the field.

6.4.4 End key

The End key moves the cursor after the last non-space character. Pressing the End key again moves the cursor to the end of the field. Repeated pressing moves the cursor back and forth.

6.4.5 Home key

The Home key moves the cursor to the first non-space character. Pressing the Home key again moves the cursor to the beginning of the field. Repeated pressing moves the cursor back and forth.

6.4.6 Insert key

The Insert key changes the insert mode.

The value of the insert mode is used in all following `ACCEPT` statements while the program is running.

When the insert mode is on, typed characters move the existing characters to the right until field is full. When it is off, typed characters type over existing characters.

Note: The insert mode is ignored for fields with a size of 1.

The insert mode can also be changed by the `COB_INSERT_MODE` setting at any time, see Appendix H [Runtime Configuration], page 70.

6.4.7 Tab keys

The Tab key returns from the `ACCEPT` with the `COB-SCR-TAB` value of 2007.

The Shift-Tab key returns with the `COB-SCR-BACK-TAB` value of 2008.

6.5 Extended `DISPLAY` statement

Extended `DISPLAY` statements allow for full control of items that display on the screen. Items display by line and column positioning.

```
DISPLAY variable-1 | literal-1 | figurative constant
  LINE line COLUMN column
  WITH BELL
    BLANK LINE | SCREEN
    ERASE EOL | EOS
    SIZE [IS] variable-2 | literal-2
END-DISPLAY.
```

6.5.1 BELL

Ring the bell. It is optional.

6.5.2 BLANK

Clear the whole line or screen. It is optional.

`BLANK LINE`

Clear the line from the beginning of the line to the end of the line.

BLANK SCREEN

Clear the whole screen.

6.5.3 ERASE

Clear the line or screen from LINE and COLUMN. It is optional.

ERASE EOL

Clear the line from LINE and COLUMN to the end of the line.

ERASE EOS

Clear the screen from LINE and COLUMN to the end of the screen.

6.5.4 SIZE

The size of *variable-1*, *literal-1*, or *figurative-constant* to display onto the screen. It is optional.

SIZE *positive-integer*

If SIZE is less than the length of *variable-1* or *literal-1* then only the SIZE number of characters display.

If SIZE is greater than the length of *variable-1* or *literal-1*, then the screen pads with spaces after the field to the SIZE length.

Figurative constants display repeatedly the number of times in SIZE. Except that LOW-VALUES always positions the cursor (see SIZE ZERO below).

SIZE ZERO

<SIZE option not specified>

Variable-1 or *literal-1* displays with the field length.

6.5.5 Figurative Constants

Certain figurative constants and values have special functions. All other figurative constants display as a single character.

SPACE Display spaces from LINE and COLUMN to the end of the screen. This is the same as WITH ERASE EOS.

LOW-VALUE

Position the cursor to LINE and COLUMN. The next DISPLAY statement does not need a LINE or COLUMN to display at that position.

ALL X"01"

Display spaces from LINE and COLUMN to the end of the line. This is the same as WITH ERASE EOL.

ALL X"02"

Clear the whole screen. This is the same as WITH BLANK SCREEN.

ALL X"07"

Ring the bell. This is the same as WITH BELL.

6.6 CONTENT-LENGTH

FUNCTION CONTENT-LENGTH returns the length of NUL byte terminated data given a pointer:

```

identification division.
program-id. zlen.
data division.
working-storage section.
01 ptr    usage pointer.
```

```

01 str    pic x(4) value z"abc".

*> Testing CONTENT-LENGTH
procedure division.

    set ptr to address of str
    display content-length(ptr)

goback.
end program hosted.

```

6.7 CONTENT-OF

FUNCTION CONTENT-OF returns an alphanumeric field given a pointer and optional length:

Data from pointer is returned as a COBOL field either by scanning for a NUL byte or using the optional length. Reference modification of result allowed.

```

identification division.
program-id. contents.
data division.
working-storage section.
01 ptr    usage pointer.
01 str    pic x(4) value z"abc".

*> Testing CONTENT-OF
procedure division.

    set ptr to address of str
    display content-of(ptr)
    display content-of(ptr, 2)
    display content-of(ptr)(2:2)

goback.
end program hosted.

```

7 System Routines

For a complete list of supported system routines, see Appendix D [System routines], page 61.

7.1 CBL_GC_GETOPT

CBL_GC_GETOPT provides the quite well-known option parser, getopt, for GnuCOBOL. The usage of this system routine is described by the following example.

```

identification division.
program-id. prog.

data division.
working-storage section.
    78 shortoptions value "jkl".

    01 longoptions.
        05 optionrecord occurs 2 times.
            10 optionname    pic x(25).
            10 has-value     pic 9.
            10 valpoint      pointer value NULL.
            10 return-value  pic x(4).

    01 longind      pic 99.
    01 long-only    pic 9 value 1.

    01 return-char  pic x(4).
    01 opt-val      pic x(10).

    01 counter      pic 9 value 0.
```

We first need to define the necessary fields for getopt's shortoptions (so), longoptions (lo), longoption index (longind), long-only-option (long-only) and also the fields for return values return-char and opt-val (arbitrary size with trimming, see return codes).

The shortoptions are written down as an alphanumeric field (i.e., a string with arbitrary size) as follows:

```
"ab:c::d"
```

This means we want getopt to look for shortoptions named a, b, c or d and we demand an option value for b and we are accepting an optional one for c.

The longoptions are defined as a table of records with oname, has-value, valpoint and val.

- oname defines the name of a longoption.
- has-value defines if an option value is demanded (has-val = 1), optional (has-val = 2) or not required (has-val = 0).
- valpoint is a pointer used to specify an address to save getopt's return value to. The pointer is optional. If it is NULL, getopt returns a value as usual. If you use the pointer it has to point to a PIC X(4) field.
- The field val is a PIC X(4) character which is returned if the longoption was recognized.

The longoption structure is immutable! You can only vary the number of records.

Now we have the tools to run CBL_GC_GETOPT within the procedure division.

```

procedure division.
    move "version" to optionname    (1).
```

```

move 0          to has-value    (1).
move "v"        to return-value (1).

move "verbose"  to optionname   (2).
move 0          to has-value    (2).
move "V"        to return-value (2).

perform with test after until return-code = -1
  call 'CBL_GC_GETOPT' using
    by reference shortoptions longoptions longind
    by value long-only
    by reference return-char opt-val
  end-call

  display return-char end-display
  display opt-val      end-display
end-perform
stop run.

```

The example shows how we initialize all parameters and call the routine until CBL_GC_GETOPT runs out of options and returns -1.

The return-char might contain the following:

- regular character if an option was recognized
- '?' if we have an undefined or ambiguous option
- '1' if we have a non-option (only if first byte of so is '-')
- '0' if valpoint != NULL and we are writing the return value to the specified address
- '-1' if we don't have any more options (or reach the first non-option if first byte of so is '+')

The return-codes of CBL_GC_GETOPT are:

- 1 if we've got a non-option (only if first byte of so is '-')
- 0 if valpoint != NULL and we are writing the return value to the specified address
- -1 if we don't have any more options (or reach the first non-option if first byte of so is '+')
- 2 if we have got an truncated option value in opt-val (because opt-val was too small)
- 3 if we got a regular answer from getopt

7.2 CBL_GC_HOSTED

CBL_GC_HOSTED provides access to the following C hosted variables:

- `argc` to binary-long by value
- `argv` to pointer to char **
- `stdin`, `stdout`, `stderr` to pointer
- `errno` giving address of errno in pointer to binary-long, use based for more direct access and conditional access to the following variables:
 - `tzname` pointer to pointer to array of two char pointers
 - `timezone` C long, will be seconds west of UTC
 - `daylight` C int, will be 1 during daylight savings

System will need to HAVE_TIMEZONE defined for these to return anything meaningful. Attempts made when they are not available return 1 from CBL_GC_HOSTED.

It returns 0 when match, 1 on failure, case matters as does length, "arg" won't match.

The usage of this system routine is described by the following example.

```
HOSTED identification division.
  program-id. hosted.
  data division.
  working-storage section.
  01 argc  usage binary-long.
  01 argv  usage pointer.

  01 stdin usage pointer.
  01 stdout usage pointer.
  01 stderr usage pointer.

  01 errno usage pointer.
  01 err   usage binary-long based.

  01 domain usage float-long value 3.0.

  01 tzname usage pointer.
  01 tznames usage pointer based.
    05 tzs usage pointer occurs 2 times.

  01 timezone  usage binary-long.
  01 daylight  usage binary-short.

*> Testing CBL_GC_HOSTED
  procedure division.
  call "CBL_GC_HOSTED" using stdin "stdin"
  display "stdin          : " stdin
  call "feof" using by value stdin
  display "feof stdin     : " return-code

  call "CBL_GC_HOSTED" using stdout "stdout"
  display "stdout         : " stdout
  call "fprintf" using by value stdout by content "Hello" & x"0a"

  call "CBL_GC_HOSTED" using stderr "stderr"
  display "stderr         : " stderr
  call "fprintf" using by value stderr by content "on err" & x"0a"

  call "CBL_GC_HOSTED" using argc "argc"
  display "argc           : " argc

  call "CBL_GC_HOSTED" using argv "argv"
  display "argv           : " argv

  call "args" using by value argc argv

  call "CBL_GC_HOSTED" using errno "errno"
  display "&errno         : " errno
```

```

set address of err to errno
display "errno          : " err
call "acos" using by value domain
display "errno after acos(3.0): " err ", EDOM is 33"

call "CBL_GC_HOSTED" using argc "arg"
display "'arg' lookup      : " return-code
call "CBL_GC_HOSTED" using null "argc"
display "null with argc    : " return-code
display "argc is still     : " argc

*> the following only returns zero if the system has HAVE_TIMEZONE set

call "CBL_GC_HOSTED" using daylight "daylight "
display "'timezone' lookup   : " return-code

if return-code not = 0
    display "system doesn't has timezone"
else

    display "timezone is      : " timezone

    call "CBL_GC_HOSTED" using daylight "daylight "
    display "'daylight' lookup    : " return-code
    display "daylight is        : " daylight

    set environment "TZ" to "PST8PDT"
    call static "tzset" returning omitted on exception continue end-call

    call "CBL_GC_HOSTED" using tzname "tzname"
    display "'tzname' lookup      : " return-code

    *> tzs(1) will point to z"PST" and tzs(2) to z"PDT"
    if return-code equal 0 and tzname not equal null then
        set address of tznames to tzname
        if tzs(1) not equal null then
            display "tzs #1          : " tzs(1)
        end-if
        if tzs(2) not equal null then
            display "tzs #2          : " tzs(2)
        end-if
    end-if

end-if

goback.
end program hosted.

```


7.3 CBL_GC_NANOSLEEP

CBL_GC_NANOSLEEP allows you to pause the program for nanoseconds. The actual precision depends on the system.

```
*> Waiting a half second
call "CBL_GC_NANOSLEEP" using "500000000" end-call

*> Waiting five seconds using compiler string catenation for readability
call "CBL_GC_NANOSLEEP" using "500" & "0000000" end-call
```

7.4 CBL_GC_FORK

CBL_GC_FORK allows you to fork the current COBOL process to a new one. The current content of the process' storage (including LOCAL-STORAGE) will be identical, any file handles get invalid in the new process, positions and file / record locks are only available to the original process.

This system routine is not available on Windows (exception: GCC on Cygwin).

Parameters: none Returns: PID (the child process gets '0' returned, the calling process gets the PID of the created children). Negative values are returned for system dependent error codes and -1 if the function is not available on the current system.

```
IDENTIFICATION DIVISION.
PROGRAM-ID. prog.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 CHILD-PID    PIC S9(9) BINARY.
01 WAIT-STS     PIC S9(9) BINARY.
PROCEDURE DIVISION.

CALL "CBL_GC_FORK" RETURNING CHILD-PID END-CALL
EVALUATE TRUE
    WHEN CHILD-PID = ZERO
        PERFORM CHILD-CODE
    WHEN CHILD-PID > ZERO
        PERFORM PARENT-CODE
    WHEN CHILD-PID = -1
        DISPLAY 'CBL_GC_FORK is not available '
            'on the current system!'
        END-DISPLAY
        PERFORM CHILD-CODE
        MOVE 0 TO CHILD-PID
        PERFORM PARENT-CODE
    WHEN OTHER
        MULTIPLY CHILD-PID BY -1 END-MULTIPLY
        DISPLAY 'CBL_GC_FORK returned system error: '
            CHILD-PID
        END-DISPLAY
END-EVALUATE

STOP RUN.

CHILD-CODE.
CALL "C$SLEEP" USING 1 END-CALL
DISPLAY "Hello, I am the child"
```

```

END-DISPLAY
MOVE 2 TO RETURN-CODE

CONTINUE.

PARENT-CODE.
  DISPLAY "Hello, I am the parent"
  END-DISPLAY
  CALL "CBL_GC_WAITPID" USING CHILD-PID RETURNING WAIT-STS
  END-CALL
  MOVE 0 TO RETURN-CODE
  EVALUATE TRUE
    WHEN WAIT-STS >= 0
      DISPLAY 'Child ended with status: '
        WAIT-STS
      END-DISPLAY
    WHEN WAIT-STS = -1
      DISPLAY 'CBL_GC_WAITPID is not available '
        'on the current system!'
      END-DISPLAY
    WHEN WAIT-STS < -1
      MULTIPLY -1 BY WAIT-STS END-MULTIPLY
      DISPLAY 'CBL_GC_WAITPID returned system error: ' WAIT-STS
      END-DISPLAY
  END-EVALUATE

CONTINUE.

```

7.5 CBL_GC_WAITPID

CBL_GC_WAITPID allows you to wait until another system process ended. Additionally you can check the process' return code.

Parameters: none Returns: function-status / child-status Negative values are returned for system dependent error codes and -1 if the function is not available on the current system.

```

CALL "CBL_GC_WAITPID" USING CHILD-PID RETURNING WAIT-STS
END-CALL
MOVE 0 TO RETURN-CODE
DISPLAY 'CBL_GC_WAITPID ended with status: ' WAIT-STS
END-DISPLAY

```

Appendix A Compiler cobc options

The following list of options was extracted from `cobc --help` and shows all available compiler options with a short description.

A.1 Options

- `-h, -help`
display this help and exit
- `-V, -version`
display compiler version and exit
- `-i, -info`
display compiler information (build/environment) and exit
- `-v, -verbose`
display compiler version and the commands invoked by the compiler
- `-vv, -verbose=2`
like `-v` but additional pass verbose option to assembler/compiler
- `-vvv, -verbose=3`
like `-vv` but additional pass verbose option to linker
- `-q, -brief`
reduced displays, commands invoked not shown
- `###`
like `-v` but commands not executed
- `-x`
build an executable program
- `-m`
build a dynamically loadable module (default)
- `-j [args], -job[=args]`
run program after build, passing *args*
- `-std=diect`
warnings/features for a specific dialect *dialect* can be one of: default, cobol2014, cobol2002, cobol85, xopen, ibm-strict, ibm, mvs-strict, mvs, mf-strict, mf, bs2000-strict, bs2000, acu-strict, acu, rm-strict, rm; see configuration files in directory config
- `-F, -free`
use free source format
- `-fixed`
use fixed source format (default)
- `-O, -O2, -O3, -Os`
enable optimization
- `-O0`
disable optimization
- `-g`
enable C compiler debug / stack check / trace
- `-d, -debug`
enable all run-time error checking
- `-o file`
place the output into *file*
- `-b`
combine all input files into a single dynamically loadable module
- `-E`
preprocess only; do not compile or link

- `-C` translation only; convert `COBOL` to `C`
- `-S` compile only; output assembly file
- `-c` compile and assemble, but do not link
- `-T file` generate and place a wide program listing into *file*
- `-t file` generate and place a program listing into *file*
- `--tlines=lines`
specify lines per page in listing, default = 55
- `-P[=dir or file]`
generate preprocessed program listing (`.lst`)
- `-Xref` generate cross reference through 'cobxref' (V. Coen's 'cobxref' must be in path)
- `-I directory`
add *directory* to copy/include search path
- `-L directory`
add *directory* to library search path
- `-l lib` link the library *lib*
- `-A options`
add *options* to the `C` compile phase
- `-Q options`
add *options* to the `C` link phase
- `-D define`
define *define* for `COBOL` compilation
- `-K entry` generate `CALL` to *entry* as static
- `-conf=file`
user-defined dialect configuration; see `-std`
- `-list-reserved`
display reserved words
- `-list-intrinsics`
display intrinsic functions
- `-list-mnemonics`
display mnemonic names
- `-list-system`
display system routines
- `-save-temps[=dir]`
save intermediate files ; default: current directory
- `-ext extension`
add file extension for resolving `COPY`

A.2 Warning options

- W enable all warnings
- Wall enable most warnings (all except as noted below)
- Wno-warning
disable warning enabled by -W or -Wall
- Wno-unfinished
do not warn if unfinished features are used ; *always active*
- Wno-pending
do not warn if pending features are mentioned ; *always active*
- Wobsolete
warn if obsolete features are used
- Warchaic
warn if archaic features are used
- Wredefinition
warn about incompatible redefinition of data items
- Wtruncate
warn about field truncation from constant assignments
- Wpossible-truncate
warn about possible field truncation ; *not set with -Wall*
- Woverlap
warn about overlapping MOVE of items
- Wpossible-overlap
warn about MOVE of items that may overlap depending on variables ; *not set with -Wall*
- Wparentheses
warn about lack of parentheses around AND within OR
- Wstrict-typing
warn strictly about type mismatch
- Wimplicit-define
warn about implicitly defined data items
- Wcorresponding
warn about CORRESPONDING with no matching items
- Winitial-value
warn if initial VALUE clause is ignored
- Wprototypes
warn about missing FUNCTION prototypes/definitions
- Warithmetic-osvs
warn if arithmetic expression precision has changed
- Wcall-params
warn about non 01/77 items for CALL parameters ; *not set with -Wall*
- Wconstant-expression
warn about expressions that always resolve to true/false

-Wcolumn-overflow
warn about text after program-text area, **FIXED** format ; *not set with -Wall*

-Wterminator
warn about lack of scope terminator **END-XXX** ; *not set with -Wall*

-Wlinkage
warn about dangling **LINKAGE** items ; *not set with -Wall*

-Wunreachable
warn about likely unreachable statements ; *not set with -Wall*

-Wno-dialect
do not warn about dialect specific issues ; *always active*

-Wothers do not warn about different issues ; *always active*

-Werror treat all warnings as errors

-Werror=warning
treat specified *warning* as error

A.3 Compiler options

-fsign=[ASCII|EBCDIC]
define display sign representation ; default: machine native

-ffold-copy=[UPPER|LOWER]
fold **COPY** subject to value ; default: no transformation

-ffold-call=[UPPER|LOWER]
fold **PROGRAM-ID**, **CALL**, **CANCEL** subject to value ; default: no transformation

-fdefaultbyte=value
initialize fields without **VALUE** to value ; decimal 0..255 or any quoted character ;
default: initialize to picture

-fmax-errors=number
maximum number of errors to report before compilation is aborted ; default: 100

-fdump=scope
dump data fields on abort, *scope* may be a combination of: **ALL**, **WS**, **LS**, **RD**, **FD**, **SC**

-fcallfh=function
use external provided **EXTFH** interface module *function* for I/O

-fintrinsics=[ALL|intrinsic function name(,name,...)]
intrinsics to be used without **FUNCTION** keyword

A.4 Compiler dialect configuration options

-freserved-words=value
use of complete/fixed reserved words

-ftab-width=1..12
set number of spaces that are assumed for tabs

-ftext-column=72..255
set right margin for source (fixed format only)

-fpic-length=number
maximum number of characters allowed in the **PICTURE** character-string

- fword-length=1..63
maximum word-length for COBOL (= programmer defined) words
- fliteral-length=*number*
maximum literal size in general
- fnumeric-literal-length=1..38
maximum numeric literal size
- fbinary-size=*value*
binary byte size - defines the allocated bytes according to PIC, may be one of: 2-4-8, 1-2-4-8, 1-8
- fbinary-byteorder=*value*
binary byte order, may be one of: native, big-endian
- fassign-clause=*value*
set way of interpreting ASSIGN
- fscreen-section-rules=*value*
which compiler's rules to apply to SCREEN SECTION item clauses
- ffilename-mapping
resolve file names at run time using environment variables.
- fpretty-display
alternate formatting of numeric fields
- fbinary-truncate
numeric truncation according to ANSI
- fcomplex-odo
allow complex OCCURS DEPENDING ON
- findirect-redefines
allow REDEFINES to other than last equal level number
- flarger-redefines-ok
allow larger REDEFINES items
- frelax-syntax-checks
allow certain syntax variations (e.g. REDEFINES position)
- frelax-level-hierarchy
allow non-matching level numbers
- fselect-working
require ASSIGN USING items to be in WORKING-STORAGE
- fsticky-linkage
LINKAGE-SECTION items remain allocated between invocations
- fmove-ibm
MOVE operates as on IBM (left to right, byte by byte)
- fperform-osvs
exit point of any currently executing perform is recognized if reached
- farithmetic-osvs
limit precision in intermediate results to precision of final result (less accurate)
- fconstant-folding
evaluate constant expressions at compile time

- fhostsign
allow hexadecimal value 'F' for NUMERIC test of signed PACKED DECIMAL field
- fprogram-name-redefinition
program names don't lead to a reserved identifier
- faccept-update
set WITH UPDATE clause as default for ACCEPT dest-item, instead of WITH NO UPDATE
- faccept-auto
set WITH AUTO clause as default for ACCEPT dest-item, instead of WITH TAB
- fconsole-is-crt
assume CONSOLE IS CRT if not set otherwise
- fno-echo-means-secure
NO-ECHO hides input with asterisks like SECURE
- fline-col-zero-default
assume the first item in a field DISPLAY goes at LINE 0 COL 0, not LINE 1 COL 1
- fdisplay-special-fig-consts
special behaviour of DISPLAY SPACE/ALL X'01'/ALL X'02'/ALL X'07'
- fbinary-comp-1
COMP-1 is a 16-bit signed integer
- fmove-non-numeric-lit-to-numeric-is-zero
imply zero in move of non-numeric literal to numeric items
- fcomment-paragraphs=*support*
comment paragraphs in IDENTIFICATION DIVISION (AUTHOR, DATE-WRITTEN, ...)
- fmemory-size-clause=*support*
MEMORY-SIZE clause
- fmultiple-file-tape-clause=*support*
MULTIPLE-FILE-TAPE clause
- flabel-records-clause=*support*
LABEL-RECORDS clause
- fvalue-of-clause=*support*
VALUE-OF clause
- fdata-records-clause=*support*
DATA-RECORDS clause
- ftop-level-occurs-clause=*support*
OCCURS clause on top-level
- fsynchronized-clause=*support*
SYNCHRONIZED clause
- fgoto-statement-without-name=*support*
GOTO statement without name
- fstop-literal-statement=*support*
STOP-literal statement
- fstop-identifier-statement=*support*
STOP-identifier statement

`-fdebugging-mode=support`
DEBUGGING MODE and debugging indicator

`-fuse-for-debugging=support`
USE FOR DEBUGGING

`-fpadding-character-clause=support`
PADDING CHARACTER clause

`-fnext-sentence-phrase=support`
NEXT SENTENCE phrase

`-flisting-statements=support`
listing-directive statements EJECT, SKIP1, SKIP2, SKIP3

`-ftitle-statement=support`
listing-directive statement TITLE

`-fentry-statement=support`
ENTRY statement

`-fmove-noninteger-to-alphanumeric=support`
move noninteger to alphanumeric

`-fmove-figurative-constant-to-numeric=support`
move figurative constants to numeric

`-fmove-figurative-space-to-numeric=support`
move figurative constant SPACE to numeric

`-fmove-figurative-quote-to-numeric=support`
move figurative constant QUOTE to numeric

`-fodo-without-to=support`
OCCURS DEPENDING ON without to

`-fsection-segments=support`
section segments

`-falter-statement=support`
ALTER statement

`-fcall-overflow=support`
OVERFLOW clause for CALL

`-fnumeric-boolean=support`
boolean literals (B'1010')

`-fhexadecimal-boolean=support`
hexadecimal-boolean literals (BX'A')

`-fnational-literals=support`
national literals (N'UTF-16 string')

`-fhexadecimal-national-literals=support`
hexadecimal-national literals (NX'265E')

`-fnational-character-literals=support`
non-standard national literals (NC'UTF-16 string')

`-fhp-octal-literals=support`
HP COBOL octal literals (%377)

`-facu-literals=support`
ACUCOBOL-GT literals (#B #O #H #X)

`-fword-continuation=support`
continuation of COBOL words

`-fnot-exception-before-exception=support`
NOT ON EXCEPTION before ON EXCEPTION

`-faccept-display-extensions=support`
extensions to ACCEPT and DISPLAY

`-frenames-uncommon-levels=support`
RENAMES of 01-, 66- and 77-level items

`-fsymbolic-constant=support`
constants defined in SPECIAL-NAMES

`-fconstant-78=support`
constant with level 78 item (note: has left to right precedence in expressions)

`-fconstant-01=support`
constant with level 01 CONSTANT AS/FROM item

`-fperform-varying-without-by=support`
PERFORM VARYING without BY phrase (implies BY 1)

`-freferance-out-of-declaratives=support`
references to sections not in DECLARATIVES from within DECLARATIVES

`-fprogram-prototypes=support`
CALL/CANCEL with program-prototype-name

`-fcall-convention-mnemonic=support`
specifying call-convention by mnemonic

`-fcall-convention-linkage=support`
specifying call-convention by WITH ... LINKAGE

`-fnumeric-value-for-edited-item=support`
numeric literals in VALUE clause of numeric-edited items

`-fincorrect-conf-sec-order=support`
incorrect order of CONFIGURATION SECTION paragraphs

`-fdefine-constant-directive=support`
allow >> DEFINE CONSTANT var AS literal

`-ffree-redefines-position=support`
REDEFINES clause not following entry-name in definition

`-frecords-mismatch-record-clause=support`
record sizes does not match RECORD clause

`-frecord-delimiter=support`
RECORD DELIMITER clause

`-fsequential-delimiters=support`
BINARY-SEQUENTIAL and LINE-SEQUENTIAL phrases in RECORD DELIMITER

`-frecord-delim-with-fixed-recs=support`
RECORD DELIMITER clause on file with fixed-length records

`-fmissing-statement=support`
 missing statement (e.g. empty IF / PERFORM)

`-fzero-length-literals=support`
 zero-length literals, e.g. " and ""

`-fxml-generate-extra-phrases=support`
 XML GENERATE's phrases other than COUNT IN

`-fcontinue-after=support`
 AFTER phrase in CONTINUE statement

`-fgoto-entry=support`
 ENTRY FOR GOTO and GOTO ENTRY statements where *support* is one of the following:
 'ok', 'warning', 'archaic', 'obsolete', 'skip', 'ignore', 'error', 'unconformable'

`-fnot-reserved=word`
 word to be taken out of the reserved words list

`-freserved=word`
 word to be added to reserved words list

`-freserved=word:alias`
 word to be added to reserved words list as alias

`-fnot-register=word`
 special register to disable

`-fregister=word`
 special register to enable

Appendix B Reserved Words

The following list of reserved words was extracted from `cobc --list-reserved` and shows the reserved words, an implementation

Please notice: This list is highly specific to the option `-std=diect` and reserved word options (`-freserved=word`, `-fno-reserved=word`) in effect. You can get the list for a given *dialect* by calling `cobc -std=diect --list-reserved`.

B.1 Common reserved words

| Reserved word | Implemented | Aliases |
|---------------------|-------------|---------|
| 3-D | Yes (C/S) | |
| ABSENT | Yes | |
| ACCEPT | Yes | |
| ACCESS | Yes | |
| ACTION | Yes (C/S) | |
| ACTIVE-CLASS | No | |
| ACTIVE-X | Yes (C/S) | |
| ADD | Yes | |
| ADDRESS | Yes | |
| ADJUSTABLE-COLUMNS | Yes (C/S) | |
| ADVANCING | Yes | |
| AFTER | Yes | |
| ALIGNED | No | |
| ALIGNMENT | Yes (C/S) | |
| ALL | Yes | |
| ALLOCATE | Yes | |
| ALLOWING | Yes (C/S) | |
| ALPHABET | Yes | |
| ALPHABETIC | Yes | |
| ALPHABETIC-LOWER | Yes | |
| ALPHABETIC-UPPER | Yes | |
| ALPHANUMERIC | Yes | |
| ALPHANUMERIC-EDITED | Yes | |
| ALSO | Yes | |
| ALTER | Yes | |
| ALTERNATE | Yes | |
| AND | Yes | |
| ANY | Yes | |
| ANYCASE | No | |
| APPLY | Yes (C/S) | |
| ARE | Yes | |
| AREA | Yes | AREAS |
| AREAS | Yes | AREA |
| ARGUMENT-NUMBER | Yes | |
| ARGUMENT-VALUE | Yes | |
| ARITHMETIC | Yes (C/S) | |
| AS | Yes | |
| ASCENDING | Yes | |
| ASCII | Yes (C/S) | |
| ASSIGN | Yes | |

| | | |
|--------------------------|-----------|--------------------------|
| AT | Yes | |
| ATTRIBUTE | Yes (C/S) | |
| ATTRIBUTES | Yes (C/S) | |
| AUTO | Yes (C/S) | AUTO-SKIP, AUTOTERMINATE |
| AUTO-DECIMAL | Yes (C/S) | |
| AUTO-SKIP | Yes | AUTO, AUTOTERMINATE |
| AUTO-SPIN | Yes (C/S) | |
| AUTOMATIC | Yes | |
| AUTOTERMINATE | Yes | AUTO, AUTO-SKIP |
| AWAY-FROM-ZERO | Yes (C/S) | |
| B-AND | No | |
| B-NOT | No | |
| B-OR | No | |
| B-XOR | No | |
| BACKGROUND-COLOR | Yes (C/S) | BACKGROUND-COLOUR |
| BACKGROUND-COLOUR | Yes | BACKGROUND-COLOR |
| BACKGROUND-HIGH | Yes | |
| BACKGROUND-LOW | Yes | |
| BACKGROUND-STANDARD | Yes | |
| BAR | Yes (C/S) | |
| BASED | Yes | |
| BEEP | Yes | BELL |
| BEFORE | Yes | |
| BELL | Yes (C/S) | BEEP |
| BINARY | Yes | |
| BINARY-C-LONG | Yes | |
| BINARY-CHAR | Yes | |
| BINARY-DOUBLE | Yes | BINARY-LONG-LONG |
| BINARY-INT | Yes | BINARY-LONG |
| BINARY-LONG | Yes | BINARY-INT |
| BINARY-LONG-LONG | Yes | BINARY-DOUBLE |
| BINARY-SEQUENTIAL | Yes (C/S) | |
| BINARY-SHORT | Yes | |
| BIT | Yes | |
| BITMAP | Yes (C/S) | |
| BITMAP-END | Yes (C/S) | |
| BITMAP-HANDLE | Yes (C/S) | |
| BITMAP-NUMBER | Yes (C/S) | |
| BITMAP-START | Yes (C/S) | |
| BITMAP-TIMER | Yes (C/S) | |
| BITMAP-TRAILING | Yes (C/S) | |
| BITMAP-TRANSPARENT-COLOR | Yes (C/S) | |
| BITMAP-WIDTH | Yes (C/S) | |
| BLANK | Yes | |
| BLINK | Yes (C/S) | |
| BLOCK | Yes | |
| BOOLEAN | No | |
| BOTTOM | Yes | |
| BOX | Yes (C/S) | |
| BOXED | Yes (C/S) | |
| BULK-ADDITION | Yes (C/S) | |
| BUSY | Yes (C/S) | |

| | | |
|-------------------|-----------|---------|
| BUTTONS | Yes (C/S) | |
| BY | Yes | |
| BYTE-LENGTH | Yes (C/S) | |
| C | Yes (C/S) | |
| CALENDAR-FONT | Yes (C/S) | |
| CALL | Yes | |
| CANCEL | Yes | |
| CANCEL-BUTTON | Yes (C/S) | |
| CAPACITY | Yes (C/S) | |
| CARD-PUNCH | Yes (C/S) | |
| CARD-READER | Yes (C/S) | |
| CASSETTE | Yes (C/S) | |
| CCOL | Yes (C/S) | |
| CD | Yes | |
| CELL | Yes (C/S) | CELLS |
| CELL-COLOR | Yes (C/S) | |
| CELL-DATA | Yes (C/S) | |
| CELL-FONT | Yes (C/S) | |
| CELL-PROTECTION | Yes (C/S) | |
| CELLS | Yes | CELL |
| CENTER | Yes (C/S) | |
| CENTERED | Yes (C/S) | |
| CENTERED-HEADINGS | Yes (C/S) | |
| CENTURY-DATE | Yes (C/S) | |
| CF | Yes | |
| CH | Yes | |
| CHAIN | No | |
| CHAINING | Yes | |
| CHARACTER | Yes | |
| CHARACTERS | Yes | |
| CHECK-BOX | Yes (C/S) | |
| CLASS | Yes | |
| CLASS-ID | No | |
| CLASSIFICATION | Yes (C/S) | |
| CLEAR-SELECTION | Yes (C/S) | |
| CLINE | Yes (C/S) | |
| CLINES | Yes (C/S) | |
| CLOSE | Yes | |
| COBOL | Yes (C/S) | |
| CODE | Yes | |
| CODE-SET | Yes | |
| COL | Yes | |
| COLLATING | Yes | |
| COLOR | Yes | |
| COLORS | Yes (C/S) | COLOURS |
| COLOURS | Yes | COLORS |
| COLS | Yes | |
| COLUMN | Yes | |
| COLUMN-COLOR | Yes (C/S) | |
| COLUMN-DIVIDERS | Yes (C/S) | |
| COLUMN-FONT | Yes (C/S) | |
| COLUMN-HEADINGS | Yes (C/S) | |

| | | |
|--------------------|-----------|-----------------|
| COLUMN-PROTECTION | Yes (C/S) | |
| COLUMNS | Yes | |
| COMBO-BOX | Yes (C/S) | |
| COMMA | Yes | |
| COMMAND-LINE | Yes | |
| COMMIT | Yes | |
| COMMON | Yes | |
| COMMUNICATION | Yes | |
| COMP | Yes | COMPUTATIONAL |
| COMP-0 | Yes | COMPUTATIONAL-0 |
| COMP-1 | Yes | COMPUTATIONAL-1 |
| COMP-2 | Yes | COMPUTATIONAL-2 |
| COMP-3 | Yes | COMPUTATIONAL-3 |
| COMP-4 | Yes | COMPUTATIONAL-4 |
| COMP-5 | Yes | COMPUTATIONAL-5 |
| COMP-6 | Yes | COMPUTATIONAL-6 |
| COMP-N | Yes | COMPUTATIONAL-N |
| COMP-X | Yes | COMPUTATIONAL-X |
| COMPUTATIONAL | Yes | COMP |
| COMPUTATIONAL-0 | Yes | COMP-0 |
| COMPUTATIONAL-1 | Yes | COMP-1 |
| COMPUTATIONAL-2 | Yes | COMP-2 |
| COMPUTATIONAL-3 | Yes | COMP-3 |
| COMPUTATIONAL-4 | Yes | COMP-4 |
| COMPUTATIONAL-5 | Yes | COMP-5 |
| COMPUTATIONAL-6 | Yes | COMP-6 |
| COMPUTATIONAL-N | Yes | COMP-N |
| COMPUTATIONAL-X | Yes | COMP-X |
| COMPUTE | Yes | |
| CONDITION | Yes | |
| CONFIGURATION | Yes | |
| CONSTANT | Yes | |
| CONTAINS | Yes | |
| CONTENT | Yes | |
| CONTINUE | Yes | |
| CONTROL | Yes | |
| CONTROLS | Yes | |
| CONVERSION | Yes (C/S) | |
| CONVERTING | Yes | |
| COPY | Yes | |
| COPY-SELECTION | Yes (C/S) | |
| CORR | Yes | CORRESPONDING |
| CORRESPONDING | Yes | CORR |
| COUNT | Yes | |
| CRT | Yes | |
| CRT-UNDER | Yes | |
| Csize | Yes (C/S) | |
| CURRENCY | Yes | |
| CURSOR | Yes | |
| CURSOR-COL | Yes (C/S) | |
| CURSOR-COLOR | Yes (C/S) | |
| CURSOR-FRAME-WIDTH | Yes (C/S) | |

| | | |
|-----------------------|-----------|------------|
| CURSOR-ROW | Yes (C/S) | |
| CURSOR-X | Yes (C/S) | |
| CURSOR-Y | Yes (C/S) | |
| CUSTOM-PRINT-TEMPLATE | Yes (C/S) | |
| CYCLE | Yes (C/S) | |
| DASHED | Yes (C/S) | |
| DATA | Yes | |
| DATA-COLUMNS | Yes (C/S) | |
| DATA-POINTER | No | |
| DATA-TYPES | Yes (C/S) | |
| DATE | Yes | |
| DATE-ENTRY | Yes (C/S) | |
| DAY | Yes | |
| DAY-OF-WEEK | Yes | |
| DE | Yes | |
| DEBUGGING | Yes | |
| DECIMAL-POINT | Yes | |
| DECLARATIVES | Yes | |
| DEFAULT | Yes | |
| DEFAULT-BUTTON | Yes (C/S) | |
| DEFAULT-FONT | Yes | |
| DELETE | Yes | |
| DELIMITED | Yes | |
| DELIMITER | Yes | |
| DEPENDING | Yes | |
| DESCENDING | Yes | |
| DESTINATION | Yes | |
| DESTROY | Yes | |
| DETAIL | Yes | |
| DISABLE | Yes | |
| DISC | Yes (C/S) | |
| DISK | Yes (C/S) | |
| DISPLAY | Yes | |
| DISPLAY-COLUMNS | Yes (C/S) | |
| DISPLAY-FORMAT | Yes (C/S) | |
| DIVIDE | Yes | |
| DIVIDER-COLOR | Yes (C/S) | |
| DIVIDERS | Yes (C/S) | |
| DIVISION | Yes | |
| DOTDASH | Yes (C/S) | |
| DOTTED | Yes (C/S) | |
| DOUBLE | Yes | FLOAT-LONG |
| DOWN | Yes | |
| DRAG-COLOR | Yes (C/S) | |
| DROP-DOWN | Yes (C/S) | |
| DROP-LIST | Yes (C/S) | |
| DUPLICATES | Yes | |
| DYNAMIC | Yes | |
| EBCDIC | Yes (C/S) | |
| EC | Yes | |
| ECHO | Yes | |
| EGI | Yes | |

| | | |
|-------------------|-----------|-------------|
| ELEMENT | Yes (C/S) | |
| ELSE | Yes | |
| EMI | Yes | |
| EMPTY-CHECK | Yes | REQUIRED |
| ENABLE | Yes | |
| ENCODING | Yes (C/S) | |
| ENCRYPTION | Yes (C/S) | |
| END | Yes | |
| END-ACCEPT | Yes | |
| END-ADD | Yes | |
| END-CALL | Yes | |
| END-CHAIN | No | |
| END-COLOR | Yes (C/S) | |
| END-COMPUTE | Yes | |
| END-DELETE | Yes | |
| END-DISPLAY | Yes | |
| END-DIVIDE | Yes | |
| END-EVALUATE | Yes | |
| END-IF | Yes | |
| END-JSON | Yes | |
| END-MODIFY | Yes (C/S) | |
| END-MULTIPLY | Yes | |
| END-OF-PAGE | Yes | EOP |
| END-PERFORM | Yes | |
| END-READ | Yes | |
| END-RECEIVE | Yes | |
| END-RETURN | Yes | |
| END-REWRITE | Yes | |
| END-SEARCH | Yes | |
| END-START | Yes | |
| END-STRING | Yes | |
| END-SUBTRACT | Yes | |
| END-UNSTRING | Yes | |
| END-WRITE | Yes | |
| END-XML | Yes | |
| ENGRAVED | Yes (C/S) | |
| ENSURE-VISIBLE | Yes (C/S) | |
| ENTRY | Yes | |
| ENTRY-CONVENTION | Yes (C/S) | |
| ENTRY-FIELD | Yes (C/S) | |
| ENTRY-REASON | Yes (C/S) | |
| ENVIRONMENT | Yes | |
| ENVIRONMENT-NAME | Yes | |
| ENVIRONMENT-VALUE | Yes | |
| EO | No | |
| EOL | Yes (C/S) | |
| EOP | Yes | END-OF-PAGE |
| EOS | Yes (C/S) | |
| EQUAL | Yes | EQUALS |
| EQUALS | Yes | EQUAL |
| ERASE | Yes (C/S) | |
| ERROR | Yes | |

| | | |
|--------------------|-----------|-------------|
| ESCAPE | Yes | |
| ESCAPE-BUTTON | Yes (C/S) | |
| ESI | Yes | |
| EVALUATE | Yes | |
| EVENT | Yes | |
| EVENT-LIST | Yes (C/S) | |
| EVERY | Yes (C/S) | |
| EXCEPTION | Yes | |
| EXCEPTION-OBJECT | No | |
| EXCEPTION-VALUE | Yes (C/S) | |
| EXCLUSIVE | Yes | |
| EXIT | Yes | |
| EXPAND | Yes (C/S) | |
| EXPANDS | No (C/S) | |
| EXTEND | Yes | |
| EXTERN | Yes (C/S) | |
| EXTERNAL | Yes | |
| EXTERNAL-FORM | Yes | |
| F | Yes (C/S) | |
| FACTORY | No | |
| FALSE | Yes | |
| FD | Yes | |
| FH--FCD | Yes (C/S) | |
| FH--KEYDEF | Yes (C/S) | |
| FILE | Yes | |
| FILE-CONTROL | Yes | |
| FILE-ID | Yes | |
| FILE-NAME | Yes (C/S) | |
| FILE-POS | Yes (C/S) | |
| FILL-COLOR | Yes (C/S) | |
| FILL-COLOR2 | Yes (C/S) | |
| FILL-PERCENT | Yes (C/S) | |
| FILLER | Yes | |
| FINAL | Yes | |
| FINISH-REASON | Yes (C/S) | |
| FIRST | Yes | |
| FIXED | Yes | |
| FIXED-FONT | Yes | |
| FIXED-WIDTH | Yes (C/S) | |
| FLAT | Yes (C/S) | |
| FLAT-BUTTONS | Yes (C/S) | |
| FLOAT | Yes | FLOAT-SHORT |
| FLOAT-BINARY-128 | No | |
| FLOAT-BINARY-32 | No | |
| FLOAT-BINARY-64 | No | |
| FLOAT-DECIMAL-16 | Yes | |
| FLOAT-DECIMAL-34 | Yes | |
| FLOAT-EXTENDED | No | |
| FLOAT-INFINITY | No | |
| FLOAT-LONG | Yes | DOUBLE |
| FLOAT-NOT-A-NUMBER | No (C/S) | |
| FLOAT-SHORT | Yes | FLOAT |

| | | |
|-----------------------|-----------|-------------------|
| FLOATING | Yes | |
| FONT | Yes | |
| FOOTING | Yes | |
| FOR | Yes | |
| FOREGROUND-COLOR | Yes (C/S) | FOREGROUND-COLOUR |
| FOREGROUND-COLOUR | Yes | FOREGROUND-COLOR |
| FOREVER | Yes (C/S) | |
| FORMAT | No | |
| FRAME | Yes (C/S) | |
| FRAMED | Yes (C/S) | |
| FREE | Yes | |
| FROM | Yes | |
| FULL | Yes (C/S) | LENGTH-CHECK |
| FULL-HEIGHT | Yes (C/S) | |
| FUNCTION | Yes | |
| FUNCTION-ID | Yes | |
| FUNCTION-POINTER | No | |
| GENERATE | Yes | |
| GET | No | |
| GIVING | Yes | |
| GLOBAL | Yes | |
| GO | Yes | |
| GO-BACK | Yes (C/S) | |
| GO-FORWARD | Yes (C/S) | |
| GO-HOME | Yes (C/S) | |
| GO-SEARCH | Yes (C/S) | |
| GOBACK | Yes | |
| GRAPHICAL | Yes (C/S) | |
| GREATER | Yes | |
| GRID | Yes (C/S) | |
| GROUP | Yes | |
| GROUP-USAGE | No | |
| GROUP-VALUE | Yes (C/S) | |
| HANDLE | Yes | |
| HAS-CHILDREN | Yes (C/S) | |
| HEADING | Yes | |
| HEADING-COLOR | Yes (C/S) | |
| HEADING-DIVIDER-COLOR | Yes (C/S) | |
| HEADING-FONT | Yes (C/S) | |
| HEAVY | Yes (C/S) | |
| HEIGHT-IN-CELLS | Yes (C/S) | |
| HIDDEN-DATA | Yes (C/S) | |
| HIGH-COLOR | Yes (C/S) | |
| HIGH-VALUE | Yes | HIGH-VALUES |
| HIGH-VALUES | Yes | HIGH-VALUE |
| HIGHLIGHT | Yes (C/S) | |
| HOT-TRACK | Yes (C/S) | |
| HSCROLL | Yes (C/S) | |
| HSCROLL-POS | Yes (C/S) | |
| I-O | Yes | |
| I-O-CONTROL | Yes | |
| ICON | Yes (C/S) | |

| | | |
|-----------------|-----------|-------------|
| ID | Yes | |
| IDENTIFICATION | Yes | |
| IDENTIFIED | Yes | |
| IF | Yes | |
| IGNORE | Yes | |
| IGNORING | Yes (C/S) | |
| IMPLEMENTS | No (C/S) | |
| IN | Yes | |
| INDEPENDENT | Yes (C/S) | |
| INDEX | Yes | |
| INDEXED | Yes | |
| INDICATE | Yes | |
| INHERITS | No | |
| INITIAL | Yes | |
| INITIALISE | Yes | INITIALIZE |
| INITIALISED | Yes | INITIALIZED |
| INITIALIZE | Yes | INITIALISE |
| INITIALIZED | Yes (C/S) | INITIALISED |
| INITIATE | Yes | |
| INPUT | Yes | |
| INPUT-OUTPUT | Yes | |
| INQUIRE | Yes | |
| INSERT-ROWS | Yes (C/S) | |
| INSERTION-INDEX | Yes (C/S) | |
| INSPECT | Yes | |
| INTERFACE | No | |
| INTERFACE-ID | No | |
| INTERMEDIATE | Yes (C/S) | |
| INTO | Yes | |
| INTRINSIC | Yes (C/S) | |
| INVALID | Yes | |
| INVOKE | No | |
| IS | Yes | |
| ITEM | Yes (C/S) | |
| ITEM-TEXT | Yes (C/S) | |
| ITEM-TO-ADD | Yes (C/S) | |
| ITEM-TO-DELETE | Yes (C/S) | |
| ITEM-TO-EMPTY | Yes (C/S) | |
| ITEM-VALUE | Yes (C/S) | |
| JSON | Yes | |
| JUST | Yes | JUSTIFIED |
| JUSTIFIED | Yes | JUST |
| KEPT | Yes | |
| KEY | Yes | |
| KEYBOARD | Yes (C/S) | |
| LABEL | Yes | |
| LABEL-OFFSET | Yes (C/S) | |
| LARGE-FONT | Yes | |
| LARGE-OFFSET | Yes (C/S) | |
| LAST | Yes | |
| LAST-ROW | Yes (C/S) | |
| LAYOUT-DATA | Yes (C/S) | |

| | | |
|-----------------|-----------|------------|
| LAYOUT-MANAGER | Yes | |
| LC_ALL | No (C/S) | |
| LC_COLLATE | No (C/S) | |
| LC_CTYPE | No (C/S) | |
| LC_MESSAGES | No (C/S) | |
| LC_MONETARY | No (C/S) | |
| LC_NUMERIC | No (C/S) | |
| LC_TIME | No (C/S) | |
| LEADING | Yes | |
| LEADING-SHIFT | Yes (C/S) | |
| LEFT | Yes | |
| LEFT-JUSTIFY | No | |
| LEFT-TEXT | Yes (C/S) | |
| LEFTLINE | Yes | |
| LENGTH | Yes | |
| LENGTH-CHECK | Yes | FULL |
| LESS | Yes | |
| LIMIT | Yes | |
| LIMITS | Yes | |
| LINAGE | Yes | |
| LINAGE-COUNTER | Yes | |
| LINE | Yes | |
| LINE-COUNTER | Yes | |
| LINE-SEQUENTIAL | Yes (C/S) | |
| LINES | Yes | |
| LINES-AT-ROOT | Yes (C/S) | |
| LINKAGE | Yes | |
| LIST-BOX | Yes (C/S) | |
| LM-RESIZE | Yes | |
| LOC | Yes (C/S) | |
| LOCAL-STORAGE | Yes | |
| LOCALE | Yes | |
| LOCK | Yes | |
| LONG-DATE | Yes (C/S) | |
| LOW-COLOR | Yes (C/S) | |
| LOW-VALUE | Yes | LOW-VALUES |
| LOW-VALUES | Yes | LOW-VALUE |
| LOWER | Yes (C/S) | |
| LOWERED | Yes (C/S) | |
| LOWLIGHT | Yes (C/S) | |
| MAGNETIC-TAPE | Yes (C/S) | |
| MANUAL | Yes | |
| MASS-UPDATE | Yes (C/S) | |
| MAX-LINES | Yes (C/S) | |
| MAX-PROGRESS | Yes (C/S) | |
| MAX-TEXT | Yes (C/S) | |
| MAX-VAL | Yes (C/S) | |
| MEDIUM-FONT | Yes | |
| MEMORY | Yes (C/S) | |
| MENU | Yes | |
| MERGE | Yes | |
| MESSAGE | Yes | |

| | | |
|------------------------|-----------|-------|
| METHOD | No | |
| METHOD-ID | No | |
| MIN-VAL | Yes (C/S) | |
| MINUS | Yes | |
| MODE | Yes | |
| MODIFY | Yes | |
| MODULES | Yes (C/S) | |
| MOVE | Yes | |
| MULTILINE | Yes (C/S) | |
| MULTIPLE | Yes | |
| MULTIPLY | Yes | |
| NAME | Yes (C/S) | |
| NAMESPACE | Yes (C/S) | |
| NAMESPACE-PREFIX | Yes (C/S) | |
| NATIONAL | Yes | |
| NATIONAL-EDITED | Yes | |
| NATIVE | Yes | |
| NAVIGATE-URL | Yes (C/S) | |
| NEAREST-AWAY-FROM-ZERO | Yes (C/S) | |
| NEAREST-EVEN | Yes (C/S) | |
| NEAREST-TOWARD-ZERO | Yes (C/S) | |
| NEGATIVE | Yes | |
| NESTED | Yes | |
| NEW | Yes | |
| NEXT | Yes | |
| NEXT-ITEM | Yes (C/S) | |
| NO | Yes | |
| NO-AUTO-DEFAULT | Yes (C/S) | |
| NO-AUTOSEL | Yes (C/S) | |
| NO-BOX | Yes (C/S) | |
| NO-DIVIDERS | Yes (C/S) | |
| NO-ECHO | Yes | |
| NO-F4 | Yes (C/S) | |
| NO-FOCUS | Yes (C/S) | |
| NO-GROUP-TAB | Yes (C/S) | |
| NO-KEY-LETTER | Yes (C/S) | |
| NO-SEARCH | Yes (C/S) | |
| NO-UPDOWN | Yes (C/S) | |
| NONE | No (C/S) | |
| NONNUMERIC | Yes (C/S) | |
| NORMAL | Yes (C/S) | |
| NOT | Yes | |
| NOTAB | Yes (C/S) | |
| NOTHING | Yes | |
| NOTIFY | Yes (C/S) | |
| NOTIFY-CHANGE | Yes (C/S) | |
| NOTIFY-DBLCLICK | Yes (C/S) | |
| NOTIFY-SELCHANGE | Yes (C/S) | |
| NULL | Yes | NULLS |
| NULLS | Yes | NULL |
| NUM-COL-HEADINGS | Yes (C/S) | |
| NUM-ROWS | Yes (C/S) | |

| | | |
|------------------|-----------|--------------|
| NUMBER | Yes | |
| NUMBERS | Yes | |
| NUMERIC | Yes | |
| NUMERIC-EDITED | Yes | |
| OBJECT | Yes | |
| OBJECT-COMPUTER | Yes | |
| OBJECT-REFERENCE | No | |
| OCCURS | Yes | |
| OF | Yes | |
| OFF | Yes | |
| OK-BUTTON | Yes (C/S) | |
| OMITTED | Yes | |
| ON | Yes | |
| ONLY | Yes | |
| OPEN | Yes | |
| OPTIONAL | Yes | |
| OPTIONS | Yes | |
| OR | Yes | |
| ORDER | Yes | |
| ORGANISATION | Yes | ORGANIZATION |
| ORGANIZATION | Yes | ORGANISATION |
| OTHER | Yes | |
| OTHERS | Yes (C/S) | |
| OUTPUT | Yes | |
| OVERFLOW | Yes | |
| OVERLAP-LEFT | Yes (C/S) | OVERLAP-TOP |
| OVERLAP-TOP | Yes (C/S) | OVERLAP-LEFT |
| OVERLINE | Yes | |
| OVERRIDE | No | |
| PACKED-DECIMAL | Yes | |
| PADDING | Yes | |
| PAGE | Yes | |
| PAGE-COUNTER | Yes | |
| PAGE-SETUP | Yes (C/S) | |
| PAGED | Yes (C/S) | |
| PARAGRAPH | Yes (C/S) | |
| PARENT | Yes (C/S) | |
| PARSE | Yes (C/S) | |
| PASCAL | Yes (C/S) | |
| PASSWORD | Yes (C/S) | |
| PERFORM | Yes | |
| PERMANENT | Yes (C/S) | |
| PF | Yes | |
| PH | Yes | |
| PHYSICAL | Yes | |
| PIC | Yes | PICTURE |
| PICTURE | Yes | PIC |
| PIXEL | Yes (C/S) | PIXELS |
| PIXELS | Yes | PIXEL |
| PLACEMENT | Yes (C/S) | |
| PLUS | Yes | |
| POINTER | Yes | |

| | | |
|-------------------|-----------|-------------------|
| POP-UP | Yes (C/S) | |
| POS | Yes | |
| POSITION | Yes | |
| POSITION-SHIFT | Yes (C/S) | |
| POSITIVE | Yes | |
| PREFIXED | No (C/S) | |
| PRESENT | Yes | |
| PREVIOUS | Yes (C/S) | |
| PRINT | Yes (C/S) | |
| PRINT-NO-PROMPT | Yes (C/S) | |
| PRINT-PREVIEW | Yes (C/S) | |
| PRINTER | Yes (C/S) | |
| PRINTER-1 | Yes (C/S) | |
| PRINTING | Yes | |
| PRIORITY | Yes | |
| PROCEDURE | Yes | |
| PROCEDURE-POINTER | Yes | PROGRAM-POINTER |
| PROCEDURES | Yes | |
| PROCEED | Yes | |
| PROCESSING | Yes (C/S) | |
| PROGRAM | Yes | |
| PROGRAM-ID | Yes | |
| PROGRAM-POINTER | Yes | PROCEDURE-POINTER |
| PROGRESS | Yes (C/S) | |
| PROHIBITED | Yes (C/S) | |
| PROMPT | Yes | |
| PROPERTIES | Yes (C/S) | |
| PROPERTY | Yes | |
| PROTECTED | Yes (C/S) | |
| PROTOTYPE | No | |
| PURGE | Yes | |
| PUSH-BUTTON | Yes (C/S) | |
| QUERY-INDEX | Yes (C/S) | |
| QUEUE | Yes | |
| QUOTE | Yes | QUOTES |
| QUOTES | Yes | QUOTE |
| RADIO-BUTTON | Yes (C/S) | |
| RAISE | Yes | |
| RAISED | Yes (C/S) | |
| RAISING | No | |
| RANDOM | Yes | |
| RD | Yes | |
| READ | Yes | |
| READ-ONLY | Yes (C/S) | |
| READERS | Yes (C/S) | |
| RECEIVE | Yes | |
| RECORD | Yes | |
| RECORD-DATA | Yes (C/S) | |
| RECORD-TO-ADD | Yes (C/S) | |
| RECORD-TO-DELETE | Yes (C/S) | |
| RECORDING | Yes | |
| RECORDS | Yes | |

| | | |
|-------------------|-----------|-------------|
| RECURSIVE | Yes (C/S) | |
| REDEFINES | Yes | |
| REEL | Yes | |
| REFERENCE | Yes | |
| REFERENCES | Yes | |
| REFRESH | Yes (C/S) | |
| REGION-COLOR | Yes (C/S) | |
| RELATION | No (C/S) | |
| RELATIVE | Yes | |
| RELEASE | Yes | |
| REMAINDER | Yes | |
| REMOVAL | Yes | |
| RENAMES | Yes | |
| REPLACE | Yes | |
| REPLACING | Yes | |
| REPORT | Yes | |
| REPORTING | Yes | |
| REPORTS | Yes | |
| REPOSITORY | Yes | |
| REQUIRED | Yes (C/S) | EMPTY-CHECK |
| RESERVE | Yes | |
| RESET | Yes | |
| RESET-GRID | Yes (C/S) | |
| RESET-LIST | Yes (C/S) | |
| RESET-TABS | Yes (C/S) | |
| RESUME | No | |
| RETRY | Yes | |
| RETURN | Yes | |
| RETURNING | Yes | |
| REVERSE | Yes | |
| REVERSE-VIDEO | Yes (C/S) | |
| REVERSED | Yes | |
| REWIND | Yes | |
| REWRITE | Yes | |
| RF | Yes | |
| RH | Yes | |
| RIGHT | Yes | |
| RIGHT-ALIGN | Yes (C/S) | |
| RIGHT-JUSTIFY | No | |
| RIMMED | Yes (C/S) | |
| ROLLBACK | Yes | |
| ROUNDED | Yes | |
| ROUNDING | Yes (C/S) | |
| ROW-COLOR | Yes (C/S) | |
| ROW-COLOR-PATTERN | Yes (C/S) | |
| ROW-DIVIDERS | Yes (C/S) | |
| ROW-FONT | Yes (C/S) | |
| ROW-HEADINGS | Yes (C/S) | |
| ROW-PROTECTION | Yes (C/S) | |
| RUN | Yes | |
| S | Yes (C/S) | |
| SAME | Yes | |

| | | |
|-------------------|-----------|--------|
| SAVE-AS | Yes (C/S) | |
| SAVE-AS-NO-PROMPT | Yes (C/S) | |
| SCREEN | Yes | |
| SCROLL | Yes (C/S) | |
| SCROLL-BAR | Yes (C/S) | |
| SD | Yes | |
| SEARCH | Yes | |
| SEARCH-OPTIONS | Yes (C/S) | |
| SEARCH-TEXT | Yes (C/S) | |
| SECONDS | Yes (C/S) | |
| SECTION | Yes | |
| SECURE | Yes (C/S) | |
| SEGMENT | Yes | |
| SEGMENT-LIMIT | Yes | |
| SELECT | Yes | |
| SELECT-ALL | Yes (C/S) | |
| SELECTION-INDEX | Yes (C/S) | |
| SELECTION-TEXT | Yes (C/S) | |
| SELF | No | |
| SELF-ACT | Yes (C/S) | |
| SEND | Yes | |
| SENTENCE | Yes | |
| SEPARATE | Yes | |
| SEPARATION | Yes (C/S) | |
| SEQUENCE | Yes | |
| SEQUENTIAL | Yes | |
| SET | Yes | |
| SHADING | Yes (C/S) | |
| SHADOW | Yes (C/S) | |
| SHARING | Yes | |
| SHORT-DATE | Yes (C/S) | |
| SHOW-LINES | Yes (C/S) | |
| SHOW-NONE | Yes (C/S) | |
| SHOW-SEL-ALWAYS | Yes (C/S) | |
| SIGN | Yes | |
| SIGNED | Yes | |
| SIGNED-INT | Yes | |
| SIGNED-LONG | Yes | |
| SIGNED-SHORT | Yes | |
| SIZE | Yes | |
| SMALL-FONT | Yes | |
| SORT | Yes | |
| SORT-MERGE | Yes | |
| SORT-ORDER | Yes (C/S) | |
| SOURCE | Yes | |
| SOURCE-COMPUTER | Yes | |
| SOURCES | No | |
| SPACE | Yes | SPACES |
| SPACE-FILL | No | |
| SPACES | Yes | SPACE |
| SPECIAL-NAMES | Yes | |
| SPINNER | Yes (C/S) | |

| | | |
|-------------------|-----------|----------------------------|
| SQUARE | Yes (C/S) | |
| STANDARD | Yes | |
| STANDARD-1 | Yes | |
| STANDARD-2 | Yes | |
| STANDARD-BINARY | Yes (C/S) | |
| STANDARD-DECIMAL | Yes (C/S) | |
| START | Yes | |
| START-X | Yes (C/S) | |
| START-Y | Yes (C/S) | |
| STATEMENT | No (C/S) | |
| STATIC | Yes (C/S) | |
| STATIC-LIST | Yes (C/S) | |
| STATUS | Yes | |
| STATUS-BAR | Yes (C/S) | |
| STATUS-TEXT | Yes (C/S) | |
| STDCALL | Yes (C/S) | |
| STEP | Yes (C/S) | |
| STOP | Yes | |
| STRING | Yes | |
| STRONG | No (C/S) | |
| STYLE | Yes (C/S) | |
| SUB-QUEUE-1 | Yes | |
| SUB-QUEUE-2 | Yes | |
| SUB-QUEUE-3 | Yes | |
| SUBTRACT | Yes | |
| SUBWINDOW | Yes | |
| SUM | Yes | |
| SUPER | No | |
| SUPPRESS | Yes | |
| SYMBOL | No (C/S) | |
| SYMBOLIC | Yes | |
| SYNC | Yes | SYNCHRONISED, SYNCHRONIZED |
| SYNCHRONISED | Yes | SYNC, SYNCHRONIZED |
| SYNCHRONIZED | Yes | SYNC, SYNCHRONISED |
| SYSTEM-DEFAULT | Yes | |
| SYSTEM-INFO | Yes (C/S) | |
| SYSTEM-OFFSET | Yes | |
| TAB | Yes (C/S) | |
| TAB-TO-ADD | Yes (C/S) | |
| TAB-TO-DELETE | Yes (C/S) | |
| TABLE | Yes | |
| TALLYING | Yes | |
| TAPE | Yes (C/S) | |
| TEMPORARY | Yes (C/S) | |
| TERMINAL-INFO | Yes (C/S) | |
| TERMINATE | Yes | |
| TERMINATION-VALUE | Yes (C/S) | |
| TEST | Yes | |
| TEXT | Yes | |
| THAN | Yes | |
| THEN | Yes | |
| THREAD | Yes | |

| | | |
|------------------|-----------|----------|
| THREADS | Yes | |
| THROUGH | Yes | THRU |
| THRU | Yes | THROUGH |
| THUMB-POSITION | Yes (C/S) | |
| TILED-HEADINGS | Yes (C/S) | |
| TIME | Yes | |
| TIME-OUT | Yes (C/S) | TIMEOUT |
| TIMEOUT | Yes | TIME-OUT |
| TIMES | Yes | |
| TITLE | Yes (C/S) | |
| TITLE-POSITION | Yes (C/S) | |
| TO | Yes | |
| TOP | Yes | |
| TOWARD-GREATER | Yes (C/S) | |
| TOWARD-LESSER | Yes (C/S) | |
| TRADITIONAL-FONT | Yes | |
| TRAILING | Yes | |
| TRAILING-SHIFT | Yes (C/S) | |
| TRAILING-SIGN | No | |
| TRANSFORM | Yes | |
| TRANSPARENT | Yes (C/S) | |
| TREE-VIEW | Yes (C/S) | |
| TRUE | Yes | |
| TRUNCATION | Yes (C/S) | |
| TYPE | Yes | |
| TYPDEF | No | |
| U | Yes (C/S) | |
| UCS-4 | Yes (C/S) | |
| UNBOUNDED | Yes (C/S) | |
| UNDERLINE | Yes (C/S) | |
| UNFRAMED | Yes (C/S) | |
| UNIT | Yes | |
| UNIVERSAL | No | |
| UNLOCK | Yes | |
| UNSIGNED | Yes | |
| UNSIGNED-INT | Yes | |
| UNSIGNED-LONG | Yes | |
| UNSIGNED-SHORT | Yes | |
| UNSORTED | Yes (C/S) | |
| UNSTRING | Yes | |
| UNTIL | Yes | |
| UP | Yes | |
| UPDATE | Yes | |
| UPDATERS | Yes (C/S) | |
| UPON | Yes | |
| UPPER | Yes (C/S) | |
| USAGE | Yes | |
| USE | Yes | |
| USE-ALT | Yes (C/S) | |
| USE-RETURN | Yes (C/S) | |
| USE-TAB | Yes (C/S) | |
| USER | Yes (C/S) | |

| | | |
|-----------------|-----------|---------------|
| USER-DEFAULT | Yes | |
| USING | Yes | |
| UTF-16 | Yes (C/S) | |
| UTF-8 | Yes (C/S) | |
| V | Yes (C/S) | |
| VAL-STATUS | No | |
| VALID | No | |
| VALIDATE | Yes | |
| VALIDATE-STATUS | No | |
| VALIDATING | Yes (C/S) | |
| VALUE | Yes | VALUES |
| VALUE-FORMAT | Yes (C/S) | |
| VALUES | Yes | VALUE |
| VARIABLE | Yes (C/S) | |
| VARIANT | Yes | |
| VARYING | Yes | |
| VERTICAL | Yes (C/S) | |
| VERY-HEAVY | Yes (C/S) | |
| VIRTUAL-WIDTH | Yes (C/S) | |
| VOLATILE | Yes | |
| VPADDING | Yes (C/S) | |
| VSCROLL | Yes (C/S) | |
| VSCROLL-BAR | Yes (C/S) | |
| VSCROLL-POS | Yes (C/S) | |
| VTOP | Yes (C/S) | |
| WAIT | Yes | |
| WEB-BROWSER | Yes (C/S) | |
| WHEN | Yes | |
| WIDTH | Yes (C/S) | |
| WIDTH-IN-CELLS | Yes (C/S) | |
| WINDOW | Yes | |
| WITH | Yes | |
| WORDS | Yes | |
| WORKING-STORAGE | Yes | |
| WRAP | Yes (C/S) | |
| WRITE | Yes | |
| WRITERS | Yes (C/S) | |
| X | Yes (C/S) | |
| XML | Yes | |
| XML-DECLARATION | Yes (C/S) | |
| Y | Yes (C/S) | |
| YYYYDDD | Yes (C/S) | |
| YYYYMMDD | Yes (C/S) | |
| ZERO | Yes | ZEROES, ZEROS |
| ZERO-FILL | No (C/S) | |
| ZEROES | Yes | ZERO, ZEROS |
| ZEROS | Yes | ZERO, ZEROES |

B.2 Extra (obsolete) context sensitive words

AUTHOR, DATE-COMPILED, DATE-MODIFIED, DATE-WRITTEN, INSTALLATION, REMARKS, SECURITY

B.3 Internal registers

| Register | Implemented | Definition |
|---------------------------|-------------|------------------------------------------------|
| 'ADDRESS OF' phrase | Yes | USAGE POINTER |
| COB-CRT-STATUS | Yes | PICTURE 9(4) USAGE DISPLAY VALUE ZERO |
| DEBUG-ITEM | Yes | PICTURE X(n) USAGE DISPLAY |
| 'LENGTH OF' phrase | Yes | CONSTANT USAGE BINARY-LONG |
| NUMBER-OF-CALL-PARAMETERS | Yes | USAGE BINARY-LONG |
| RETURN-CODE | Yes | GLOBAL USAGE BINARY-LONG VALUE ZERO |
| SORT-RETURN | Yes | GLOBAL USAGE BINARY-LONG VALUE ZERO |
| TALLY | Yes | GLOBAL PICTURE 9(5) USAGE BINARY VALUE ZERO |
| WHEN-COMPILED | Yes | CONSTANT PICTURE X(16) USAGE DISPLAY |
| XML-CODE | Yes | GLOBAL PICTURE S9(9) USAGE BINARY VALUE 0 |
| JSON-CODE | Yes | GLOBAL PICTURE S9(9) USAGE BINARY VALUE 0 |

Appendix C Intrinsic Functions

The following list of intrinsic functions was extracted from `cobc --list-intrinsics` and shows the names of the available functions, an implementation note and the number of parameters.

| Intrinsic | Function | Implemented |
|---------------------------|----------|-------------|
| Parameters | | |
| ABS | Yes | 1 |
| ACOS | Yes | 1 |
| ANNUITY | Yes | 2 |
| ASIN | Yes | 1 |
| ATAN | Yes | 1 |
| BOOLEAN-OF-INTEGER | No | 2 |
| BYTE-LENGTH | Yes | 1 - 2 |
| CHAR | Yes | 1 |
| CHAR-NATIONAL | No | 1 |
| COMBINED-DATETIME | Yes | 2 |
| CONCATENATE | Yes | Unlimited |
| CONTENT-LENGTH | Yes | 1 |
| CONTENT-OF | Yes | 1 - 2 |
| COS | Yes | 1 |
| CURRENCY-SYMBOL | Yes | 0 |
| CURRENT-DATE | Yes | 0 |
| DATE-OF-INTEGER | Yes | 1 |
| DATE-TO-YYYYMMDD | Yes | 1 - 3 |
| DAY-OF-INTEGER | Yes | 1 |
| DAY-TO-YYYYDDD | Yes | 1 - 3 |
| DISPLAY-OF | No | 1 - 2 |
| E | Yes | 0 |
| EXCEPTION-FILE | Yes | 0 |
| EXCEPTION-FILE-N | No | 0 |
| EXCEPTION-LOCATION | Yes | 0 |
| EXCEPTION-LOCATION-N | No | 0 |
| EXCEPTION-STATEMENT | Yes | 0 |
| EXCEPTION-STATUS | Yes | 0 |
| EXP | Yes | 1 |
| EXP10 | Yes | 1 |
| FACTORIAL | Yes | 1 |
| FORMATTED-CURRENT-DATE | Yes | 1 |
| FORMATTED-DATE | Yes | 2 |
| FORMATTED-DATETIME | Yes | 4 - 5 |
| FORMATTED-TIME | Yes | 3 - 4 |
| FRACTION-PART | Yes | 1 |
| HIGHEST-ALGEBRAIC | Yes | 1 |
| INTEGER | Yes | 1 |
| INTEGER-OF-BOOLEAN | No | 1 |
| INTEGER-OF-DATE | Yes | 1 |
| INTEGER-OF-DAY | Yes | 1 |
| INTEGER-OF-FORMATTED-DATE | Yes | 2 |
| INTEGER-PART | Yes | 1 |
| LENGTH | Yes | 1 - 2 |

LENGTH-AN Yes 1
LOCALE-COMPARE Yes 2 - 3
LOCALE-DATE Yes 1 - 2
LOCALE-TIME Yes 1 - 2
LOCALE-TIME-FROM-SECONDS Yes 1 - 2
LOG Yes 1
LOG10 Yes 1
LOWER-CASE Yes 1
LOWEST-ALGEBRAIC Yes 1
MAX Yes Unlimited
MEAN Yes Unlimited
MEDIAN Yes Unlimited
MIDRANGE Yes Unlimited
MIN Yes Unlimited
MOD Yes 2
MODULE-CALLER-ID Yes 0
MODULE-DATE Yes 0
MODULE-FORMATTED-DATE Yes 0
MODULE-ID Yes 0
MODULE-PATH Yes 0
MODULE-SOURCE Yes 0
MODULE-TIME Yes 0
MONETARY-DECIMAL-POINT Yes 0
MONETARY-THOUSANDS-SEPARATOR Yes 0
NATIONAL-OF No 1 - 2
NUMERIC-DECIMAL-POINT Yes 0
NUMERIC-THOUSANDS-SEPARATOR Yes 0
NUMVAL Yes 1
NUMVAL-C Yes 2
NUMVAL-F Yes 1
ORD Yes 1
ORD-MAX Yes Unlimited
ORD-MIN Yes Unlimited
PI Yes 0
PRESENT-VALUE Yes Unlimited
RANDOM Yes 0 - 1
RANGE Yes Unlimited
REM Yes 2
REVERSE Yes 1
SECONDS-FROM-FORMATTED-TIME Yes 2
SECONDS-PAST-MIDNIGHT Yes 0
SIGN Yes 1
SIN Yes 1
SQRT Yes 1
STANDARD-COMPARE No 2 - 4
STANDARD-DEVIATION Yes Unlimited
STORED-CHAR-LENGTH Yes 1

SUBSTITUTE Yes Unlimited
SUBSTITUTE-CASE Yes Unlimited
SUM Yes Unlimited
TAN Yes 1
TEST-DATE-YYYYMMDD Yes 1
TEST-DAY-YYYYDDD Yes 1
TEST-FORMATTED-DATETIME Yes 2
TEST-NUMVAL Yes 1
TEST-NUMVAL-C Yes 2
TEST-NUMVAL-F Yes 1
TRIM Yes 1 - 2
UPPER-CASE Yes 1
VARIANCE Yes Unlimited
WHEN-COMPILED Yes 0
YEAR-TO-YYYY Yes 1 - 3

Appendix D System routines

The following list of system routines was extracted from `cobc --list-system` and shows the names of the available system routines along with the number of parameters.

| System routine | Parameters |
|----------------------|------------|
| SYSTEM | 1 |
| CBL_AND | 3 |
| CBL_ALARM_SOUND | 0 |
| CBL_BELL_SOUND | 0 |
| CBL_CHANGE_DIR | 1 |
| CBL_CHECK_FILE_EXIST | 2 |
| CBL_CLOSE_FILE | 1 |
| CBL_COPY_FILE | 2 |
| CBL_CREATE_DIR | 1 |
| CBL_CREATE_FILE | 5 |
| CBL_DELETE_DIR | 1 |
| CBL_DELETE_FILE | 1 |
| CBL_EQ | 3 |
| CBL_ERROR_PROC | 2 |
| CBL_EXIT_PROC | 2 |
| CBL_FLUSH_FILE | 1 |
| CBL_GET_CSR_POS | 1 |
| CBL_GET_CURRENT_DIR | 3 |
| CBL_GET_SCR_SIZE | 2 |
| CBL_IMP | 3 |
| CBL_NIMP | 3 |
| CBL_NOR | 3 |
| CBL_NOT | 2 |
| CBL_OPEN_FILE | 5 |
| CBL_OR | 3 |
| CBL_READ_FILE | 5 |
| CBL_READ_KBD_CHAR | 1 |
| CBL_RENAME_FILE | 2 |
| CBL_SET_CSR_POS | 1 |
| CBL_TOLOWER | 2 |
| CBL_Toupper | 2 |
| CBL_WRITE_FILE | 5 |
| CBL_XOR | 3 |
| CBL_GC_FORK | 0 |
| CBL_GC_GETOPT | 6 |
| CBL_GC_HOSTED | 2 |
| CBL_GC_NANOSLEEP | 1 |
| CBL_GC_PRINTABLE | 1 - 2 |
| CBL_GC_WAITPID | 1 |
| CBL_OC_GETOPT | 6 |
| CBL_OC_HOSTED | 2 |
| CBL_OC_NANOSLEEP | 1 |
| C\$CALLED BY | 1 |
| C\$CHDIR | 2 |
| C\$COPY | 3 |
| C\$DELETE | 2 |

| | |
|--------------|-------|
| C\$FILEINFO | 2 |
| C\$GETPID | 0 |
| C\$JUSTIFY | 1 - 2 |
| C\$MAKEDIR | 1 |
| C\$NARG | 1 |
| C\$PARAMSIZE | 1 |
| C\$PRINTABLE | 1 - 2 |
| C\$SLEEP | 1 |
| C\$TOLOWER | 2 |
| C\$TOUPPER | 2 |
| EXTFH | 2 |
| X"91" | 3 |
| X"E4" | 0 |
| X"E5" | 0 |
| X"F4" | 2 |
| X"F5" | 2 |

Appendix E System names

The following list of system names was extracted from `cobc --list-mnemonics` and shows the system names categorized by their type.

E.1 System names: device

SYSIN, SYSIPT, STDIN, SYSOUT, SYSLIST, SYSLST, SYSPCH, SYSPUNCH, STDOUT, PRINT, PRINTER, PRINTER-1, SYSERR, STDERR, CONSOLE

E.2 System names: feature

C01, C02, C03, C04, C05, C06, C07, C08, C09, C10, C11, C12, S01, S02, S03, S04, S05, CSP, FORMFEED, TOP, CALL-CONVENTION

E.3 System names: switch

SWITCH-0, SWITCH-1, SWITCH-2, SWITCH-3, SWITCH-4, SWITCH-5, SWITCH-6, SWITCH-7, SWITCH-8, SWITCH-9, SWITCH-10, SWITCH-11, SWITCH-12, SWITCH-13, SWITCH-14, SWITCH-15, SWITCH-16, SWITCH-17, SWITCH-18, SWITCH-19, SWITCH-20, SWITCH-21, SWITCH-22, SWITCH-23, SWITCH-24, SWITCH-25, SWITCH-26, SWITCH-27, SWITCH-28, SWITCH-29, SWITCH-30, SWITCH-31, SWITCH-32, SWITCH-33, SWITCH-34, SWITCH-35, SWITCH-36

Appendix F Compiler Configuration

The following list was extracted from `config/default.conf`.

```
# Value: any string
name: "GnuCOBOL"

# Value: enum
standard-define                                0
#      CB_STD_OC = 0,
#      CB_STD_MF,
#      CB_STD_IBM,
#      CB_STD_MVS,
#      CB_STD_BS2000,
#      CB_STD_ACU,
#      CB_STD_85,
#      CB_STD_2002,
#      CB_STD_2014

# Value: int
tab-width:                                     8
text-column:                                  72
# Maximum word-length for COBOL words / Programmer defined words
# Be aware that GC checks the word length against COB_MAX_WORDLEN
# first (currently 63)
word-length:                                  63

# Maximum literal size in general
literal-length:                                8191

# Maximum numeric literal size (absolute maximum: 38)
numeric-literal-length:                        38

# Maximum number of characters allowed in the character-string (max. 255)
pic-length:                                    255

# Value: 'mf', 'ibm'
#
assign-clause:                                 mf

# If yes, file names are resolved at run time using
# environment variables.
# For example, given ASSIGN TO "DATAFILE", the file name will be
# 1. the value of environment variable 'DD_DATAFILE' or
# 2. the value of environment variable 'dd_DATAFILE' or
# 3. the value of environment variable 'DATAFILE' or
# 4. the literal "DATAFILE"
# If no, the value of the assign clause is the file name.
#
filename-mapping:                              yes
```

```

# Alternate formatting of numeric fields
pretty-display:                yes

# Allow complex OCCURS DEPENDING ON
complex-odo:                    no

# Allow REDEFINES to other than last equal level number
indirect-redefines:            no

# Binary byte size - defines the allocated bytes according to PIC
# Value:      signed  unsigned  bytes
#            -----  -
# '2-4-8'      1 - 4      same      2
#              5 - 9      same      4
#              10 - 18     same      8
#
# '1-2-4-8'    1 - 2      same      1
#              3 - 4      same      2
#              5 - 9      same      4
#              10 - 18     same      8
#
# '1--8'       1 - 2      1 - 2      1
#              3 - 4      3 - 4      2
#              5 - 6      5 - 7      3
#              7 - 9      8 - 9      4
#              10 - 11     10 - 12     5
#              12 - 14     13 - 14     6
#              15 - 16     15 - 16     7
#              17 - 18     17 - 18     8
#
binary-size:                    1-2-4-8

# Numeric truncation according to ANSI
binary-truncate:                yes

# Binary byte order
# Value: 'native', 'big-endian'
binary-byteorder:                big-endian

# Allow larger REDEFINES items
larger-redefines-ok:            no

# Allow certain syntax variations (eg. REDEFINES position)
relax-syntax-checks:            no

# Perform type OSVS - If yes, the exit point of any currently
# executing perform is recognized if reached.
perform-osvs:                    no

# Compute intermediate decimal results like IBM OSVS
arithmetic-osvs:                no

```

```

# MOVE like IBM (mvc); left to right, byte by byte
move-ibm:                                no

# SELECT RELATIVE KEY and ASSIGN fields must be in WORKING-STORAGE
select-working:                          no

# If yes, linkage-section items remain allocated
# between invocations.
sticky-linkage:                          no

# If yes, allow non-matching level numbers
relax-level-hierarchy:                   no

# If yes, evaluate constant expressions at compile time
constant-folding:                        yes

# Allow Hex 'F' for NUMERIC test of signed PACKED DECIMAL field
hostsign:                                no

# If yes, set WITH UPDATE clause as default for ACCEPT dest-item,
# except if WITH NO UPDATE clause is used
accept-update:                           no

# If yes, set WITH AUTO clause as default for ACCEPT dest-item,
# except if WITH TAB clause is used
accept-auto:                             no

# If yes, DISPLAYs and ACCEPTs are, by default, done on the CRT (i.e., using
# curses).
console-is-crt:                          no

# If yes, allow redefinition of the current program's name. This prevents its
# use in a prototype-format CALL/CANCEL statement.
program-name-redefinition:               yes

# If yes, NO ECHO/NO-ECHO/OFF is the same as SECURE (hiding input with
# asterisks, not spaces).
no-echo-means-secure:                    no

# If yes, the first item in a field screen ACCEPT/DISPLAY (e.g. DISPLAY x UPON
# CRT) is located after the previous ACCEPT/DISPLAY (as though LINE 0 COL 0 had
# been specified).
line-col-zero-default:                   yes

# If yes, DISPLAY SPACES acts as ERASE EOS, DISPLAY X"01" acts as ERASE EOL,
# DISPLAY X"02" acts as BLANK SCREEN and DISPLAY X"07" acts as BELL. Note
# DISPLAY LOW-VALUE is excluded from this; it will always just position the
# cursor.
display-special-fig-consts:              no

# If yes, COMP-1 is a signed 16-bit integer and any PICTURE clause is ignored.
binary-comp-1:                           no

```

```
# auto-adjust to zero like MicroFocus does
move-non-numeric-lit-to-numeric-is-zero: no

# What rules to apply to SCREEN SECTION items clauses
screen-section-rules: gc

# Dialect features
# Value: 'ok', 'warning', 'archaic', 'obsolete', 'skip', 'ignore', 'error',
# 'unconformable'

alter-statement: obsolete
comment-paragraphs: obsolete
call-overflow: archaic
data-records-clause: obsolete
debugging-mode: ok
use-for-debugging: ok
listing-statements: skip # may be a user-defined word
title-statement: skip # may be a user-defined word
entry-statement: ok
goto-statement-without-name: obsolete
label-records-clause: obsolete
memory-size-clause: obsolete
move-noninteger-to-alphanumeric: error
move-figurative-constant-to-numeric: archaic
move-figurative-space-to-numeric: error
move-figurative-quote-to-numeric: obsolete
multiple-file-tape-clause: obsolete
next-sentence-phrase: archaic
odo-without-to: warning
padding-character-clause: obsolete
section-segments: ignore
stop-literal-statement: obsolete
stop-identifier-statement: obsolete
synchronized-clause: ok
top-level-occurs-clause: ok
value-of-clause: obsolete
numeric-boolean: ok
hexadecimal-boolean: ok
national-literals: ok
hexadecimal-national-literals: ok
national-character-literals: warning

acu-literals: unconformable
hp-octal-literals: unconformable
word-continuation: warning
not-exception-before-exception: ok
accept-display-extensions: ok
renames-uncommon-levels: ok
symbolic-constant: ok
constant-78: ok
constant-01: ok
```



```

perform-varying-without-by:          ok
reference-out-of-declaratives:       warning
program-prototypes:                 ok
call-convention-mnemonic:           ok
call-convention-linkage:            ok
numeric-value-for-edited-item:       ok
incorrect-conf-sec-order:            ok
define-constant-directive:          archaic
free-redefines-position:            warning
records-mismatch-record-clause      warning
record-delimiter:                   ok
sequential-delimiters:              ok
record-delim-with-fixed-recs:       ok
missing-statement:                  warning
zero-length-literals:              ok
xml-generate-extra-phrases:         ok
continue-after:                    ok
goto-entry:                        warning

```

```

# use complete word list; synonyms and exceptions are specified below
reserved-words:          default

```

```

# not-reserved:

```

```

# Value: Word to be taken out of the reserved words list

```

```

not-reserved:  TERMINAL

```

```

# reserved:

```

```

# Entries of the form word-1=word-2 define word-1 as an alias for default

```

```

# reserved word word-2. No spaces are allowed around the equal sign.

```

```

reserved:      AUTO-SKIP=AUTO
reserved:      AUTOTERMINATE=AUTO
reserved:      BACKGROUND-COLOUR=BACKGROUND-COLOR
reserved:      BEEP=BELL
reserved:      BINARY-INT=BINARY-LONG
reserved:      BINARY-LONG-LONG=BINARY-DOUBLE
reserved:      CELLS=CELL
reserved:      COLOURS=COLORS
reserved:      EMPTY-CHECK=REQUIRED
reserved:      EQUALS=EQUAL
reserved:      FOREGROUND-COLOUR=FOREGROUND-COLOR
reserved:      HIGH-VALUES=HIGH-VALUE
reserved:      INITIALISE=INITIALIZE
reserved:      INITIALISED=INITIALIZED
reserved:      LENGTH-CHECK=FULL
reserved:      LOW-VALUES=LOW-VALUE
reserved:      ORGANISATION=ORGANIZATION
reserved:      PIXELS=PIXEL
reserved:      SYNCHRONISED=SYNCHRONIZED
reserved:      TIMEOUT=TIME-OUT
reserved:      VALUES=VALUE
reserved:      ZEROES=ZERO
reserved:      ZEROS=ZERO

```

Appendix G Module loader cobcrun options

The following list of options was extracted from `cobcrun --help` and shows all available options for the module loader with a short description.

- `-h, -help`
display this help and exit
- `-V, -version`
display cobcrun and runtime version and exit
- `-i, -info`
display runtime information (build/environment)
- `-c file, -config=file`
set runtime configuration from *file*
- `-r, -runtime-config`
display current runtime configuration (value and origin for all settings)
- `-M module, -module=module`
set entry point module name and/or load path where `-M module` prepends any directory to the dynamic link loader library search path and any basename to the module preload list (`COB_LIBRARY_PATH` and/or `COB_PRELOAD`)

Appendix H Runtime configuration

The following list was extracted from `config/runtime.cfg`.

H.1 General instructions

The initial `runtime.cfg` file is found in the `$COB_CONFIG_DIR`, which defaults to `installdir/gnucobol/config` (see `cobcrun --info` for the local path that is configured). The environment variable `COB_RUNTIME_CONFIG` may define a different runtime configuration file to read.

If settings are included in the runtime environment file multiple times then the last setting value is used, no warning occurs.

Settings via environment variables always take precedence over settings that are given in runtime configuration files. And the environment is checked after completing processing of the runtime configuration file(s)

All values set to string variables or environment variables are checked for `${envvar}` and replacement is done at the time of the setting. You can also specify a default value for the case that `envvar` is not set: `${envvar:default}` (the format `${envvar:-default}` is supported, too).

Any environment variable may be set with the directive `setenv`.

Example: `setenv COB_LIBARAY_PATH ${LD_LIBRARY_PATH}`

Any environment variable may be unset with the directive `unsetenv` (one var per line).

Example: `unsetenv COB_LIBRARAY_PATH`

Runtime configuration files can include other files with the directive `include`.

Example: `include my-runtime-configuration-file`

To include another configuration file only if it is present use the directive `includeif`. You can also use `${envvar}` inside this.

Example: `includeif ${HOME}/mygc.cfg`

If you want to reset a parameter to its default value use `reset parametername`.

Most runtime variables have boolean values, some are switches, some have string values, integer values (if not explicit noted: unsigned) and some are size values. The boolean values will be evaluated as following: to true: 1, Y, ON, YES, TRUE (no matter of case) to false: 0, N, OFF A 'size' value is an unsigned integer optionally followed by K, M, or G for kilo, mega or giga.

For convenience a parameter in the `runtime.cfg` file may be defined by using either the environment variable name or the parameter name. In most cases the environment variable name is the parameter name (in upper case) with the prefix `COB_`.

For a complete list of the settings in use see `cobcrun --runtime-config`.

Note: If you want to *slightly* speed up a program's startup time, remove all of the comments from the actual real configuration file that is processed.

H.2 General environment

```
Environment name: COB_DISABLE_WARNINGS
Parameter name:  disable_warnings
Purpose:         turn off runtime warning messages
Type:            boolean
```

```

        Default: false
        Example: DISABLE_WARNINGS TRUE

Environment name: COB_ENV_MANGLE
  Parameter name: env_mangle
    Purpose: names checked in the environment would get non alphanumeric
             change to '_'
    Type: boolean
    Default: false
    Example: ENV_MANGLE TRUE

Environment name: COB_SET_DEBUG
  Parameter name: debugging_mode
    Purpose: to enable USE ON DEBUGGING procedures that were active
             during compile-time because of WITH DEBUGGING MODE,
             otherwise the code generated will be skipped
    Type: boolean
    Default: false
    Example: COB_SET_DEBUG 1

Environment name: COB_SET_TRACE
  Parameter name: set_trace
    Purpose: to enable COBOL trace feature
    Type: boolean
    Default: false
    Example: SET_TRACE TRUE

Environment name: COB_TRACE_FILE
  Parameter name: trace_file
    Purpose: to define where COBOL trace output should go
    Type: string : $$ is replaced by process id
    Default: stderr
    Example: TRACE_FILE ${HOME}/mytrace.$$

Environment name: COB_TRACE_FORMAT
  Parameter name: trace_format
    Purpose: to define format of COBOL trace output
    Type: string
    Default: "%P %S Line: %L"
             %P is replaced by Program-Id/Function-Id minimal length 29
             with prefix
             %I is replaced by Program-Id/Function-Id variable length,
             without prefix
             %L is replaced by Line number, right justified, length 6
             %S is replaced by statement type and name
             %F is replaced by source file name
    Example: TRACE_FORMAT "Line: %L %S"
    Note: format of GC2.2 and older:
          "PROGRAM-ID: %I Line: %L %S"

Environment name: COB_DUMP_FILE
  Parameter name: dump_file

```

Purpose: to define where COBOL dump output should go
 Note: The -fdump=all compile option prepares for dump
 Type: string : \$\$ is replaced by process id
 Default: stderr
 Example: DUMP_FILE \${HOME}/mytrace.log

Environment name: COB_DUMP_WIDTH
 Parameter name: dump_width
 Purpose: to define COBOL dump line length
 Type: integer
 Default: 100
 Example: dump_width 120

Environment name: COB_CURRENT_DATE
 Parameter name: current_date
 Purpose: specify an alternate Date/Time to be returned to ACCEPT clauses this is used for testing purposes or to tweak a missing offset partial setting is allowed
 Type: numeric string in format YYYYDDMMHH24MISS or date string
 Default: the operating system date is used
 Example: COB_CURRENT_DATE "2016/03/16 16:40:52"
 current_date YYYYMMDDHHMMSS+01:00

H.3 Call environment

Environment name: COB_LIBRARY_PATH
 Parameter name: library_path
 Purpose: paths for dynamically-loadable modules
 Type: string
 Note: the default paths ./installpath/extras are always added to the given paths
 Example: LIBRARY_PATH /opt/myapp/test:/opt/myapp/production

Environment name: COB_PRE_LOAD
 Parameter name: pre_load
 Purpose: modules that are loaded during startup, can be used to CALL COBOL programs or C functions that are part of a module library
 Type: string
 Note: the modules listed should NOT include extensions, the runtime will use the right ones on the various platforms, COB_LIBRARY_PATH is used to locate the modules
 Example: PRE_LOAD COBOL_function_library:external_c_library

Environment name: COB_LOAD_CASE
 Parameter name: load_case
 Purpose: resolve ALL called program names to UPPER or LOWER case
 Type: Only use UPPER or LOWER

Default: if not set program names in CALL are case sensitive
 Example: LOAD_CASE UPPER

Environment name: COB_PHYSICAL_CANCEL

Parameter name: physical_cancel

Purpose: physically unload a dynamically-loadable module on CANCEL, this frees some RAM and allows the change of modules during run-time but needs more time to resolve CALLs (both to active and not-active programs)

Alias: default_cancel_mode, LOGICAL_CANCEL (0 = yes)

Type: boolean (evaluated for true only)

Default: false

Example: PHYSICAL_CANCEL TRUE

H.4 File I/O

Environment name: COB_VARSEQ_FORMAT

Parameter name: varseq_format

Purpose: declare format used for variable length sequential files
 - different types and lengths precede each record
 - 'length' is the data length, does not include the prefix

Type: 0 means 2 byte record length (big-endian) + 2 NULs
 1 means 4 byte record length (big-endian)
 2 means 4 byte record length (local machine int)
 3 means 2 byte record length (big-endian)

Default: 0

Example: VARSEQ_FORMAT 1

Environment name: COB_FILE_PATH

Parameter name: file_path

Purpose: define default location where data files are stored

Type: file path directory

Default: . (current directory)

Example: FILE_PATH \${HOME}/mydata

Environment name: COB_LS_FIXED

Parameter name: ls_fixed

Purpose: Defines if LINE SEQUENTIAL files should be fixed length (or variable, by removing trailing spaces)

Alias: STRIP_TRAILING_SPACES (0 = yes)

Type: boolean

Default: false

Example: LS_FIXED TRUE

Environment name: COB_LS_NULLS

Parameter name: ls_nulls

Purpose: Defines for LINE SEQUENTIAL files what to do with data which is not DISPLAY type. This could happen if a LINE

SEQUENTIAL record has COMP data fields in it.
 Type: boolean
 Default: false
 Note: The TRUE setting will handle files that contain COMP data in a similar manner to the method used by Micro Focus
 Example: LS_NULL = TRUE

Environment name: COB_SYNC
 Parameter name: sync
 Purpose: Should the file be synced to disk after each write/update
 Type: boolean
 Default: false
 Example: SYNC: TRUE

Environment name: COB_SORT_MEMORY
 Parameter name: sort_memory
 Purpose: Defines how much RAM to assign for sorting data if this size is exceeded the SORT will be done on disk instead of memory
 Type: size but must be more than 1M
 Default: 128M
 Example: SORT_MEMORY 64M

Environment name: COB_SORT_CHUNK
 Parameter name: sort_chunk
 Purpose: Defines how much RAM to assign for sorting data in chunks
 Type: size but must be within 128K and 16M
 Default: 256K
 Example: SORT_CHUNK 1M

H.5 Screen I/O

Environment name: COB_BELL
 Parameter name: bell
 Purpose: Defines how a request for the screen to beep is handled
 Type: FLASH, SPEAKER, FALSE, BEEP
 Default: BEEP
 Example: BELL SPEAKER

Environment name: COB_REDIRECT_DISPLAY
 Parameter name: redirect_display
 Purpose: Defines if DISPLAY output should be sent to 'stderr'
 Type: boolean
 Default: false
 Example: redirect_display Yes

Environment name: COB_SCREEN_ESC
 Parameter name: screen_esc

Purpose: Enable handling of ESC key during ACCEPT
 Type: boolean
 Default: false
 Note: is only evaluated if COB_SCREEN_EXCEPTIONS is active
 Example: screen_esc Yes

Environment name: COB_SCREEN_EXCEPTIONS
 Parameter name: screen_exceptions
 Purpose: enable exceptions for function keys during ACCEPT
 Type: boolean
 Default: false
 Example: screen_exceptions Yes

Environment name: COB_TIMEOUT_SCALE
 Parameter name: timeout_scale
 Purpose: specify translation in milliseconds for ACCEPT clauses
 BEFORE TIME value / AFTER TIMEOUT
 Type: integer
 0 means 1000 (Micro Focus COBOL compatible), 1 means 100
 (ACUCOBOL compatible), 2 means 10, 3 means 1
 Default: 0
 Note: the minimum and possible maximum value depend on the
 screenio library used
 Example: timeout_scale 3

Environment name: COB_INSERT_MODE
 Parameter name: insert_mode
 Purpose: specify default insert mode for ACCEPT; 0=off, 1=on
 Type: boolean
 Default: false
 Note: also sets the cursor type (if available)
 Example: insert_mode Y

Environment name: COB_MOUSE_FLAGS
 Parameter name: mouse_flags
 Purpose: specify which mouse events will be sent as function key
 to the application during ACCEPT and how they will be
 handled
 Type: int (by bits)
 Default: 1
 Note: 0 disables the mouse cursor, any other value enables it,
 any value containing 1 will enable internal handling (click
 to position, double-click to enter).
 See copy/screenio.cpy for list of events and their values.
 Alias: MOUSE_FLAGS
 Example: 11 (enable internal handling => 1, left press => 2,
 double-click => 8; 1+2+8=11)

Environment name: COB_MOUSE_INTERVAL
 Parameter name: mouse_interval
 Purpose: specifies the maximum time (in thousands of a second)
 that can elapse between press and release events for them

to be recognized as a click.

Type: int (0 - 166)

Default: 100

Note: 0 disables the click resolution (instead press + release are recognized), also disables positioning by mouse click

Environment name: COB_DISPLAY_PRINT_PIPE

Parameter name: display_print_pipe

Purpose: Defines command line used for sending output of DISPLAY UPON PRINTER to (via pipe)
This is very similar to Micro Focus COBPRINTER

Note: Each executed DISPLAY UPON PRINTER statement causes a new invocation of command-line (= new process start).
Each invocation receives the data referenced in the DISPLAY statement and is followed by an end-of-file condition.
COB_DISPLAY_PRINT_FILE, if set, takes precedence over COB_DISPLAY_PRINT_PIPE.

Alias: COBPRINTER

Type: string

Default: not set

Example: print 'cat >>/tmp/myprt.log'

Environment name: COB_DISPLAY_PRINT_FILE

Parameter name: display_print_file

Purpose: Defines file to be appended to by DISPLAY UPON PRINTER

Note: Each DISPLAY UPON PRINTER opens, appends and closes the file.

Type: string : \$\$ is replaced by process id

Default: not set

Example: display_printer '/tmp/myprt.log'

Environment name: COB_DISPLAY_PUNCH_FILE

Parameter name: display_punch_file

Purpose: Defines file to be created on first DISPLAY UPON SYSPUNCH/SYSPCH

Note: The file will be only be closed on runtime exit.

Type: string : \$\$ is replaced by process id

Default: not set

Example: display_punch './punch_\$\$out'

Environment name: COB_LEGACY

Parameter name: legacy

Purpose: keep behavior of former runtime versions, currently only for setting screen attributes for non input fields

Type: boolean

Default: not set

Example: legacy true

Environment name: COB_EXIT_WAIT

Parameter name: exit_wait

Purpose: to wait on main program exit if an extended screenio DISPLAY was issued without an ACCEPT following

Type: boolean
Default: true
Example: COB_EXIT_WAIT off

Environment name: COB_EXIT_MSG

Parameter name: exit_msg

Purpose: string to display if COB_EXIT_WAIT is processed, set to ''
if no actual display but an ACCEPT should be done

Type: string
Default: 'end of program, please press a key to exit' (localized)
Example: COB_EXIT_MSG ''

H.6 Report I/O

Environment name: COB_COL_JUST_LRC

Parameter name: col_just_lrc

Purpose: If true, then COLUMN defined as LEFT, RIGHT or CENTER
will have the data justified within the field limits
If false, then the data is just copied into the column as is

Type: boolean
Default: TRUE
Example: col_just_lrc True

Appendix I GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.

<https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both

covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its

Title Page, then add an item describing the Modified Version as stated in the previous sentence.

- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled “Acknowledgements” or “Dedications”, Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled “Endorsements”. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled “Endorsements” or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <https://www.gnu.org/licenses/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C)  year  your name.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled ‘‘GNU
Free Documentation License’’.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being list their titles, with
the Front-Cover Texts being list, and with the Back-Cover Texts
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.