

# Howto build and run libgcj/gcj snapshots or cvs

1. [Get a GCC snapshot](#) or [obtain GCC via CVS](#).

2. Make a compile directory

```
$ mkdir compile
```

3. Move the snapshot into the compile dir, e.g.

```
$ cd compile
$ mv ../gcc-20001211.tar.gz .
$ gunzip *.gz
$ tar xfv *.tar
$ ln -s gcc-20001211 gcc
```

4. Tell the build system you want to build libgcj

Have a look at the toplevel configure.in (gcc/configure.in) and make sure that the variable 'noconfigdirs' isn't assigned to something (like target-libjava or \${libgcj}).)

Also, check for platform-specific assignments of 'noconfigdirs' with \${libgcj}; if \${libgcj} is listed for your platform, remove it before configuring.

5. Compile and install gcc/gcj/libgcj

```
$ cd compile
$ mkdir objdir
$ cd objdir
$ ../gcc/configure --enable-threads=posix --prefix=/home/joerg/gcc \
--enable-shared --enable-languages=c++,java \
--with-as=/opt/gnu/bin/as --with-ld=/opt/gnu/bin/ld
$ make bootstrap
$ make
$ make install
```

The Boehm GC is the collector picked up by default.

If you compile under Linux you could omit the last two options. Under Solaris you'll need them. If you omit '--prefix' the compiled source will be installed under /usr/local. For more information about installing gcc and/or configuration options see:

<http://gcc.gnu.org/install/index.html>

If you have a broken gas/bin-utils (such as Debian potato) then you want to edit the auto-host.h file and remove the definition of HAVE\_GAS\_HIDDEN after configuring, but before typeing make. See [the FAQ](#) for more information.



[GCJ Home](#)  
[GCC Home](#)  
[Status](#)  
[FAQ](#)  
[Documentation](#)  
[Contributing](#)  
[Done with GCJ](#)

## About GCC

[Mission Statement](#)  
[Releases](#)  
[Snapshots](#)  
[Mailing lists](#)  
[Contributors](#)  
[Steering Committee](#)

## Documentation

[Installation](#)  
 · [Platforms](#)  
 · [Testing](#)  
[Manual](#)  
[FAQ](#)  
[Wiki](#)  
[Further Readings](#)

## Download

[Mirror sites](#)  
[Binaries](#)

## "Live" Sources

[SVN read access](#)  
[Rsync read access](#)  
[SVN write access](#)

## Development

[Development Plan](#)  
 · [Tentative Timeline](#)  
[Contributing](#)  
[Why contribute?](#)  
[Open projects](#)  
[Front ends](#)  
[Back ends](#)

## 6. Adjust your environment

Reflect your choice of `--prefix` value to your environment. It depends on which shell you're running. For `csh` compatible shells, edit a file `env.csh`:

```
setenv PATH /home/joerg/gcc/bin:$PATH
setenv LD_LIBRARY_PATH /home/joerg/gcc/lib

$ source env.csh
```

## 7. Edit a file HelloWorld.java

```
public class HelloWorld {
    public static void main(String [] args) {
        System.out.println("Hello");
    }
}
```

## 8. Compile and run HelloWorld

```
$ gcj --main=HelloWorld -o HelloWorld HelloWorld.java
$ ./HelloWorld
```

[Extensions](#)  
[Benchmarks](#)

### Bugs

[Known bugs](#)  
[How to report](#)  
[Bug database](#)  
· [Management](#)

Please send FSF & GNU inquiries & questions to [gnu@gnu.org](mailto:gnu@gnu.org). There are also [other ways to contact](#) the FSF.

These pages are maintained by [the GCC team](#).

For questions related to the use of GCC, please consult these web pages and the [GCC manuals](#). If that fails, the [gcc-help@gcc.gnu.org](mailto:gcc-help@gcc.gnu.org) mailing list might help.

Please send comments on these web pages and the development of GCC to our developer mailing list at [gcc@gnu.org](mailto:gcc@gnu.org) or [gcc@gcc.gnu.org](mailto:gcc@gcc.gnu.org). All of our lists have [public archives](#).

Copyright (C) Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110, USA.

Verbatim copying and distribution of this entire article is permitted in any medium, provided this notice is preserved.

Last modified 2006-06-21

