

The following directions provide details on installing required tools and processes required to acquire current GCL code and compile it to run on Win32 platforms from WinXP to Win7.^{1,2} This should also work on Win8 but these directions have not been verified with Win8.

1. INSTALL AND CONFIGURE GIT FOR WINDOWS

1.1 Acquire Git for Windows from the following URL:

<http://code.google.com/p/msysgit/downloads/list?q=full+installer+official+git>

1.2 Click to choose the most current 'Full Installer...' to your computer (figure 1)

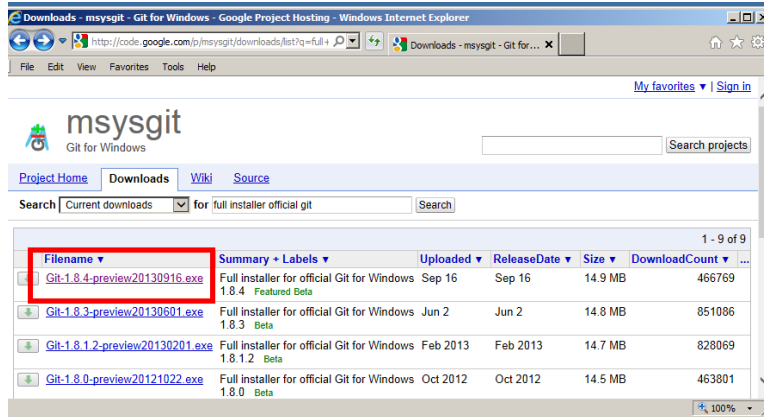


Figure 1. Choose the most current Git

1.3 Click the chosen Git version for downloading (figure 2)

1.4 Choose 'Run' in the dialog box that appears at the bottom of your browser screen (figure 2)

NOTE: If your browser alerts you that the publisher of this version can't be verified, select the Run button again to initiate the installation without copying the file to your hard drive.

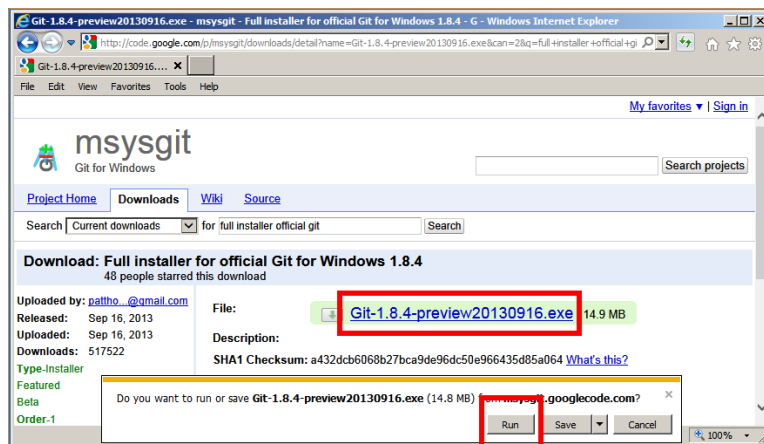


Figure 2. Choose version for downloading & run installation program

- Setup and configuration directions prepared by Don Winiecki <dwinniecki@boisestate.edu> 7. November 2013 using details provided by Camm Maguire, Mike Thomas and Gordon Novak Jr. <<http://lists.gnu.org/archive/html/axiom-developer/2003-10/msg00227.html>>
- The preferred build system for Win32 GCL is MinGW & MSYS. These directions will result in installation of the following components (as of 5. November 2013).
 - GNU C Compiler (GCC) version 4.8.1-4
 - MinGW binutils version 2.23.2-1
 - MSYS Core version 1.0.18-1-msys—1.0.18

NOTE: If a 'User Account Control' dialog box asks for confirmation to allow the program to make changes to your computer, click 'Yes' to continue

1.5 Click the 'Next' button on the 'Git Setup' dialog box (figure 3)

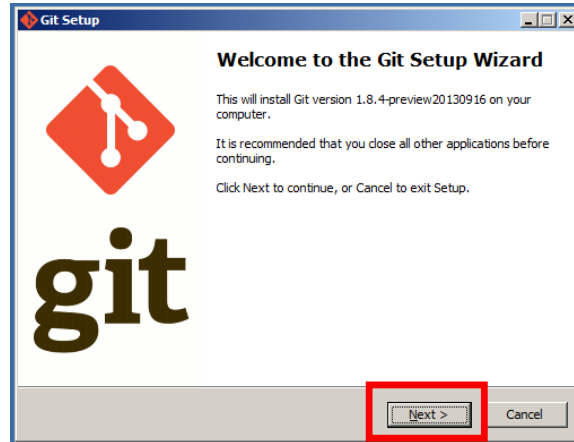


Figure 3. Beginning installation of Git

1.6 Click 'Next' to accept the GNU General Public License required to install Git

1.7 Click 'Next' to accept default components in the 'Git Setup' dialog box

1.8 Click 'Next' to accept default configuration for *adjusting your PATH settings*

1.9 Click 'Next' to accept default configuration for *Configuring the line ending conversions*

The installation process will proceed automatically to install Git on your computer. An icon labeled 'Git Bash' (figure 4) will be installed on your computer's desktop and in your Start menu.

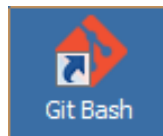


Figure 4. 'Git Bash' icon

1.10 When the installation is completed, you can choose to read the release notes, otherwise, uncheck the checkbox labeled "View ReleaseNotes.rtf" and click the 'Finish' button to complete this process.

2. INSTALL BUILD TOOLS FOR WIN32 GCL (MinGW, MSYS)³

2.1 Acquire the MinGW installer from the following URL (figure 5):

<http://sourceforge.net/projects/mingw/files/>

³ Thanks to Camm Maguire for prototyping this process.

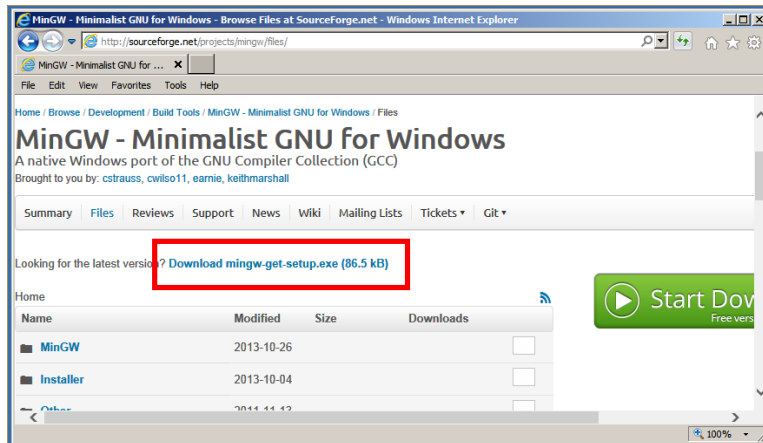


Figure 5. Select most recent MinGW setup program

2.2 When prompted, choose 'Run' from the browser prompt (figure 6)

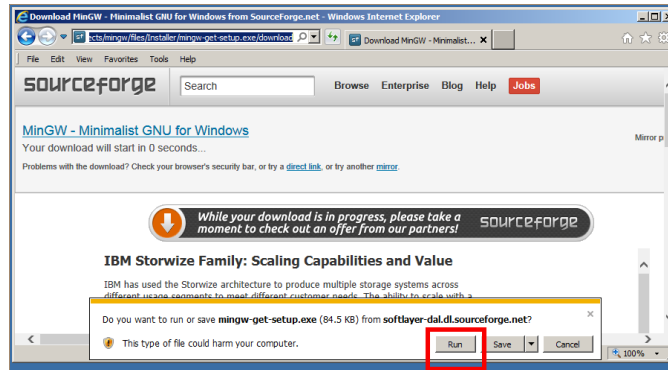


Figure 6. Running the MinGW setup program

NOTE: If your browser prompts you that the publisher of 'mingw-get-setup.exe' couldn't be verified, click 'Run' to continue.

2.3 Click 'Install' in the 'MinGW Installation Manager Setup Tool' (figure 7).

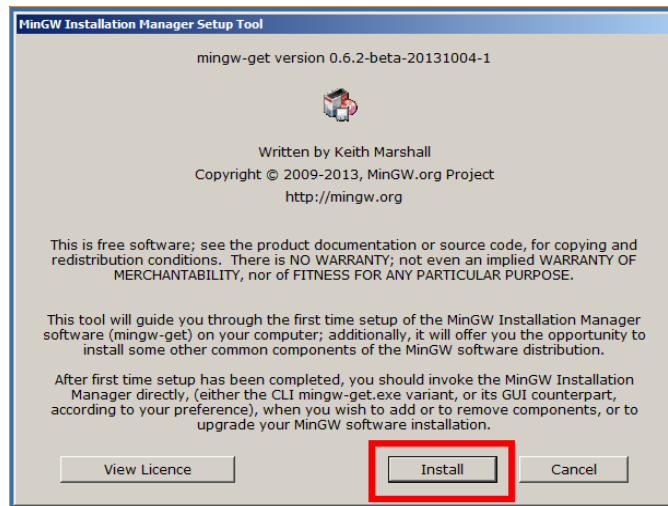


Figure 7. Initiating installation of MinGW

NOTE: The default directory for MinGW is always `c:\MinGW` (figure 8).
 If you have an existing MinGW and MSYS setup, it is recommended that you rename its directories so that the current MinGW system can install in its preferred default location.

For example, I have an older installation of MinGW and MSYS in directories at the root level of my hard drive. I renamed them `c:\~MinGW` and `c:\~MSYS` to facilitate use of the newest MinGW.

If you have an existing installation of MinGW in default directories and do not rename those directories, the following will replace that installation.

2.4 Click 'Continue' to accept default location for MinGW and installation features (figure 8).

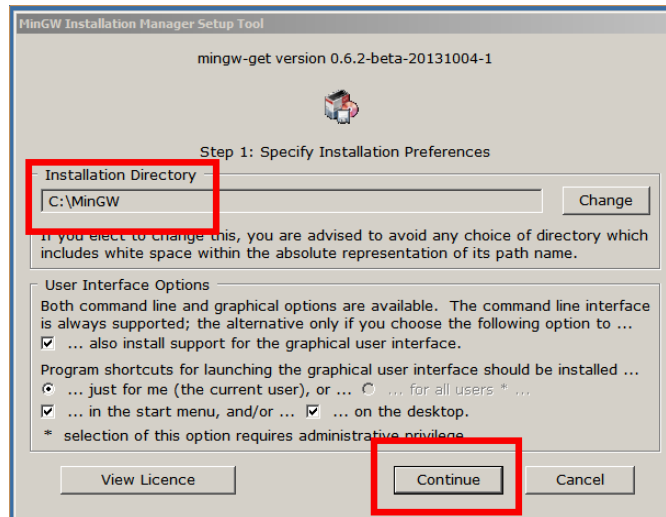


Figure 8. Installation details for MinGW

2.5 A dialog box will show progress of installation of the 'MinGW Installation Manager'

2.6 When the process is completed, click the 'Continue' button to proceed. The 'MinGW Installation Manager' window will appear (figure 9).

2.7 Click the 'Basic Setup' item in the left-most pane of the 'MinGW Installation Manager' window (figure 9)

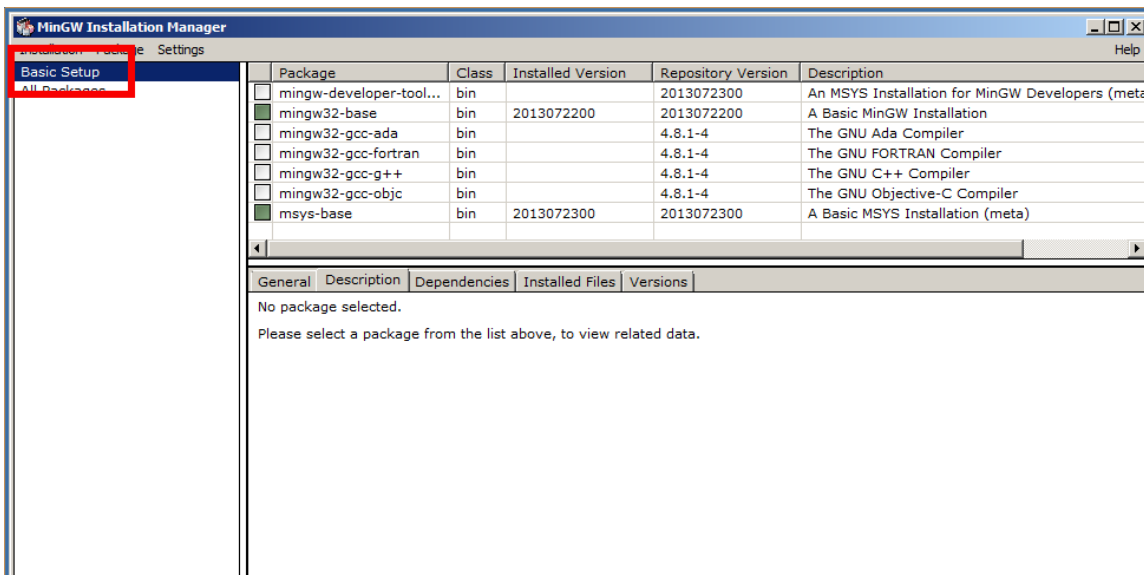


Figure 9. Basic Setup in MinGW Installation Manager

- 2.8 Click on 'mingw32-base' in the upper-right pane in the 'MinGW Installation Manager' window. In the drop-down menu that appears, select 'Mark for Installation' (figure 10).

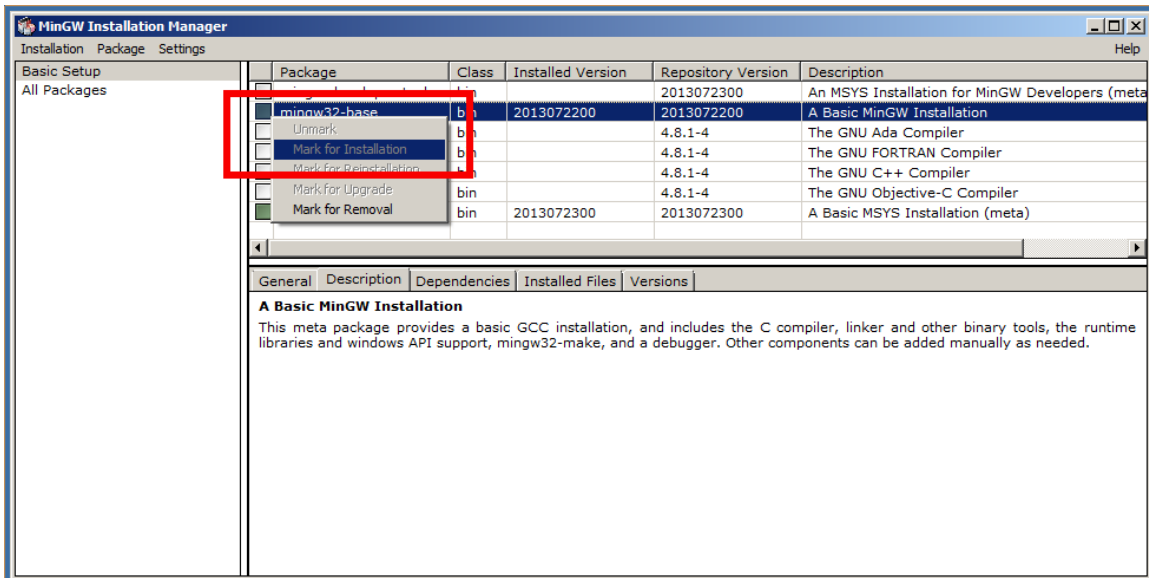


Figure 10. Selecting and Marking items for installation in 'MinGW Installation Manager' window.

- 2.9 Repeat step 2.8 (above) for 'msys-base'.
- 2.10 Click 'All Packages' and then 'MinGW' in the upper-left pane of the 'MinGW Installation Manager' window (figure 11).

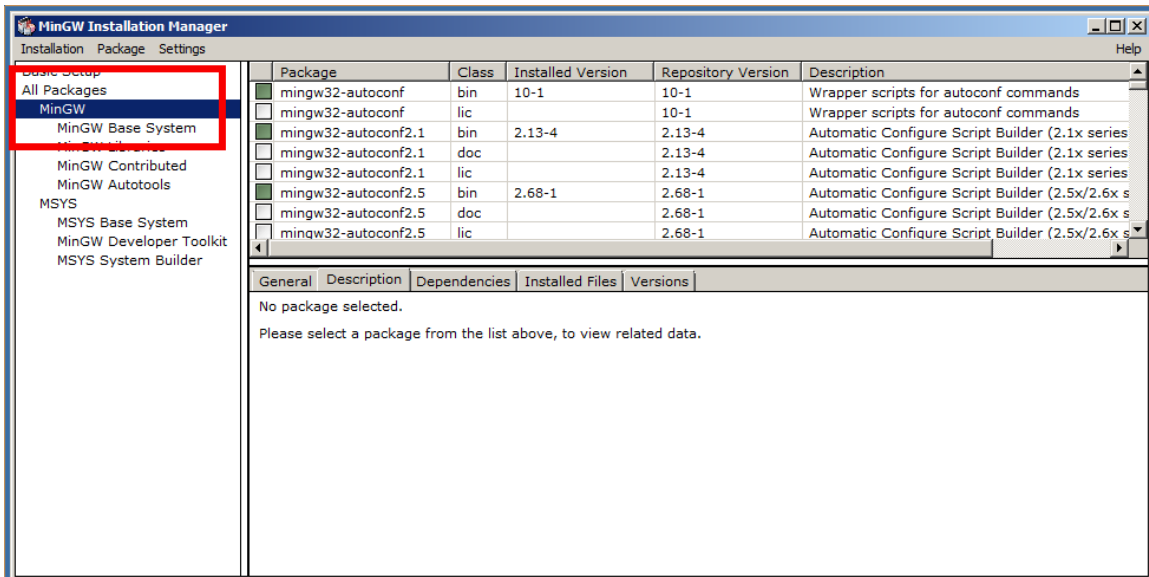


Figure 11. Selecting 'All Packages' and 'MinGW' in the 'MinGW Installation Manager' window.

- 2.11 Click on 'mingw-autoconf' ('bin' class only) in the upper-right pane in the 'MinGW Installation Manager' window. In the drop-down menu that appears, select 'Mark for Installation' (figure 12).

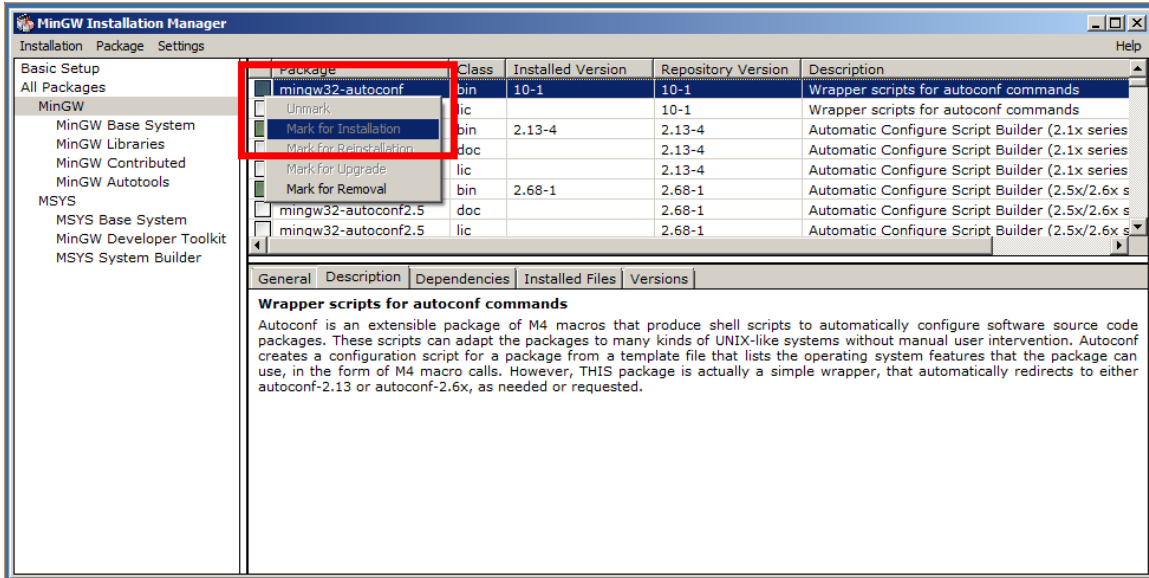


Figure 12. Selecting and Marking items for installation in 'MinGW Installation Manager' window.

2.12 Select the 'Installation' menu and choose the 'Apply Changes' menu item (figure 13).

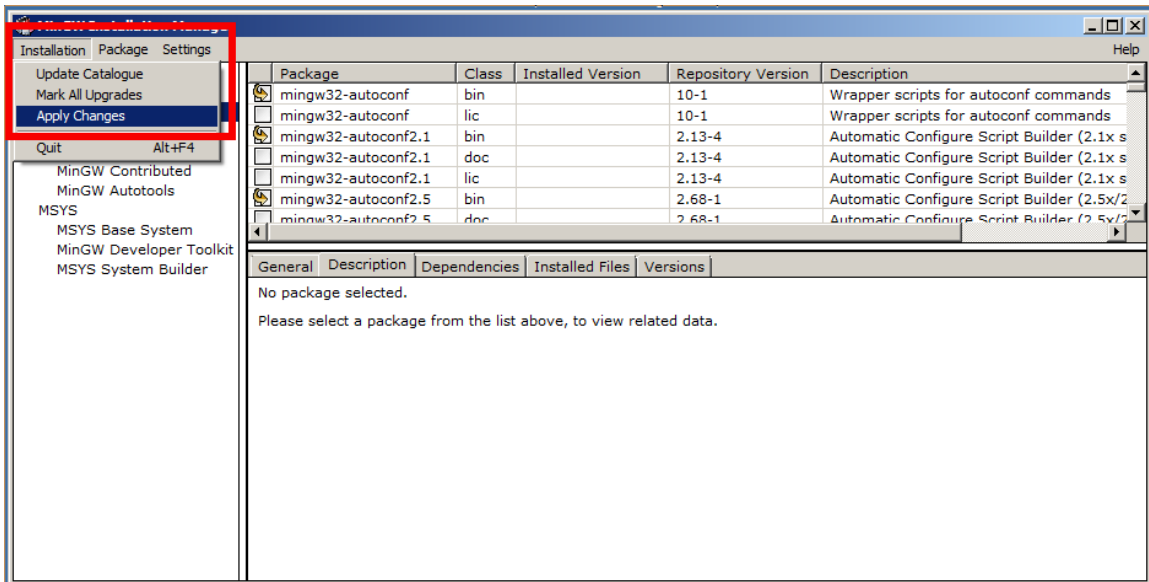


Figure 13. Preparing to install MinGW build tools for GCL.

2.13 The 'Schedule of Pending Actions' modal dialog box appears. Click the 'Apply' button in this dialog box.

Packages chosen and applied in steps 2.7 through 2.13 (above) will be downloaded and installed. The process usually required about five minutes during writing of these directions.

When the process is completed, the 'Close' button in the 'Applying Selected Changes' dialog box will become active. Click the 'Close' button to finish installation (figure 14).

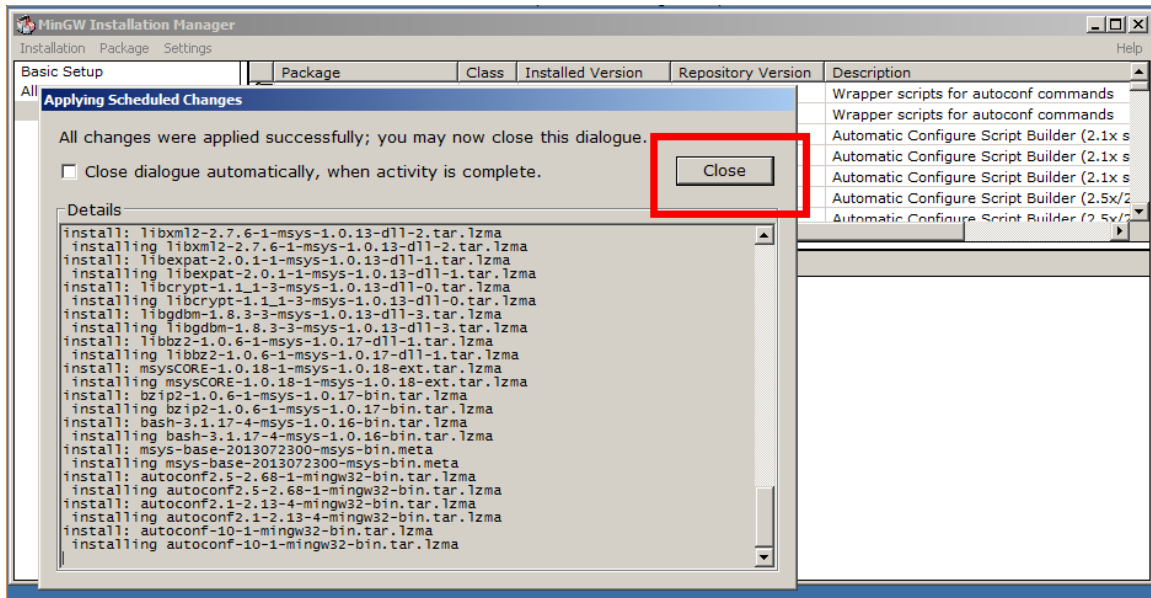


Figure 14. Close the 'Applying Scheduled Changes' dialog box.

2.14 Close the 'MinGW Installation Manager' window.

2.15 Create a shortcut to the file named 'msys.bat' in the following directory: `c:\MinGW\msys\1.0\` and move that shortcut to your desktop. This will allow you to launch the MSYS shell.

NOTE: You can assign the icon in the file named
'`c:\MinGW\msys\1.0\msys.ico`' to this shortcut.
This may make it easier to identify the shortcut on your desktop.

2.16 Edit PATH environment variable to make MinGW tools available. The method for accomplishing this varies across operating systems. Follow directions below specific to your computer's operating system.

Windows 8

I don't yet have access to a Win8 machine and have not been able to verify the process for setting the PATH on that OS. RSN...

In the meantime, please refer to the following URL, inserting

`;c:\MinGW\bin;c:\mingw\msys\1.0\bin;` for what is offered in step 3 of the directions at the following URL.

<http://computertutorialsonline.com/change-path-environment-variable-in-windows-8>

Windows 7

1. Click the 'Start' menu button and select 'Computer' on the right-side of the Start Menu.
2. Choose 'System Properties' from the window's context menu (figure 15).

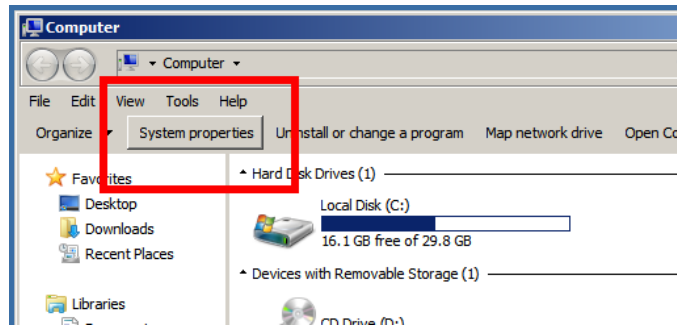


Figure 15. Choose 'System Properties'

3. Click Advanced system settings in the dialog box that appears (figure 16).

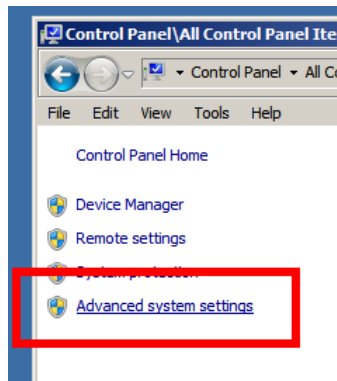


Figure 16. Choose 'Advanced system settings'

4. Choose the 'Advanced' tab in the 'System Properties' dialog box then click the 'Environment Variables...' button (figure 17).

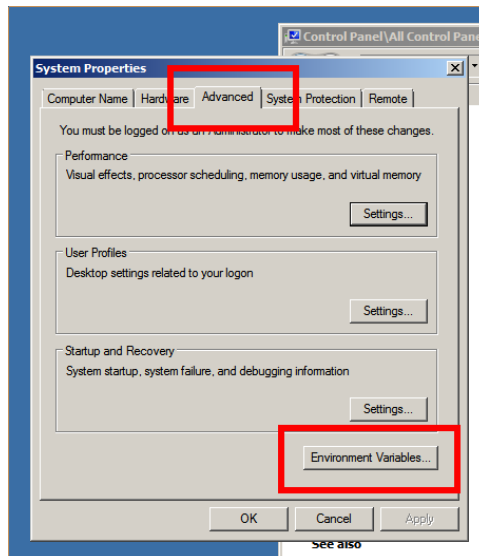


Figure 17. Accessing 'Environment Variables...' settings.

5. Click on **PATH** in the 'Environment Variables' window, then choose the 'Edit' button (figure 18).

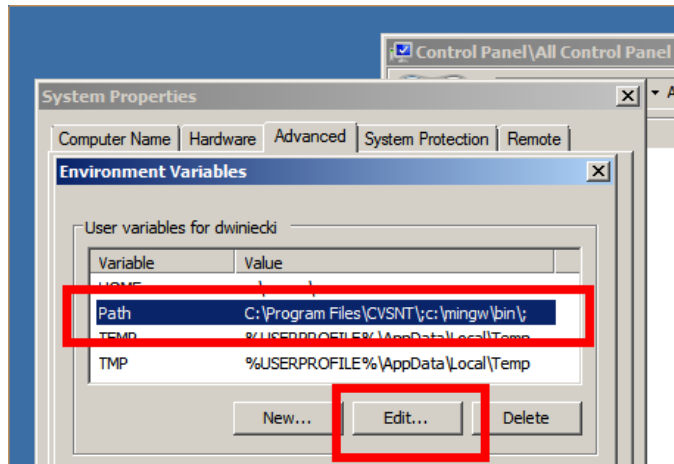


Figure 18. Accessing the PATH Environment Variable.

6. In the 'Edit User Variable' dialog box (figure 19), add the following text.

NOTE: The Variable Value should go at the *end* of the Path string.
Semi-colons are required.

Variable Name: Path

Variable Value: ;c:\MinGW\bin;c:\mingw\msys\1.0\bin;

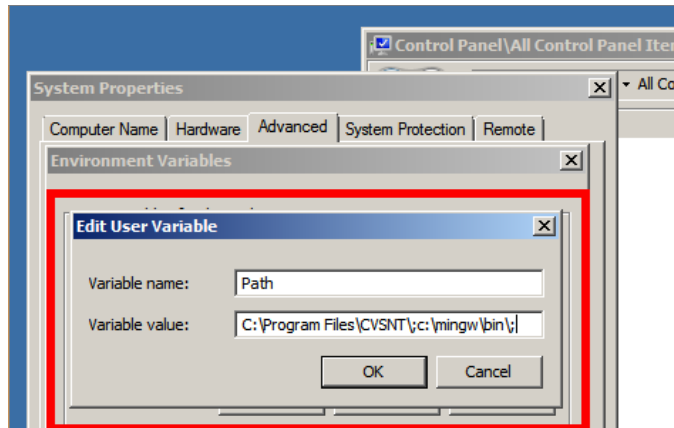


Figure 19. Adding required PATH details in the 'Edit User Variable' dialog box.

7. Close each of the opened dialog boxes as indicated in table 1 below.

Table 1. Closing Dialog Boxes After Setting Win7 Path

Dialog Box Title	Close By Doing This
Edit User Variable (figure 19)	Click the 'OK' button
Environment Variables (figure 18)	Click the 'OK' button
System Properties (figure 17)	Click the 'OK' button
Control Panel\All Control Panel Items\System (figure 16)	Click the Close Box ('X' button in the window's title bar)

Windows XP

1. Click the 'Start' menu button and select 'Control Panel' on the right-side of the Start Menu.
2. Double-click the 'System' control panel item (figure 20).

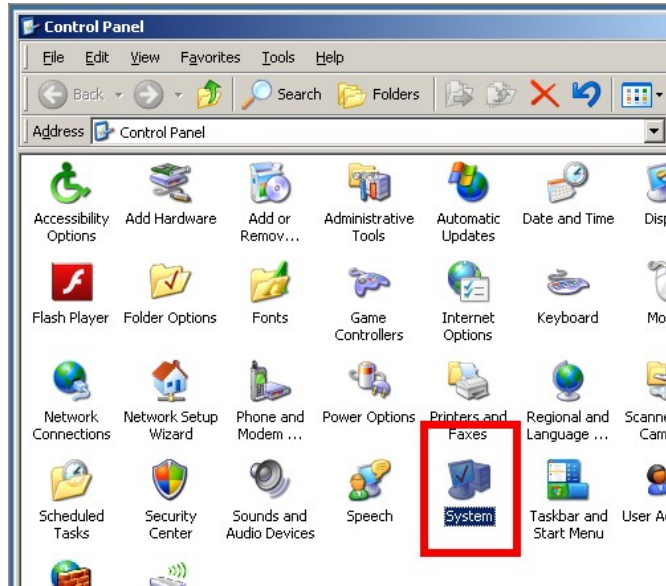


Figure 20. The 'System' control panel.

3. Click the 'Advanced' tab in the 'System Properties' dialog box, then click the 'Environment Properties' button (figure 21).

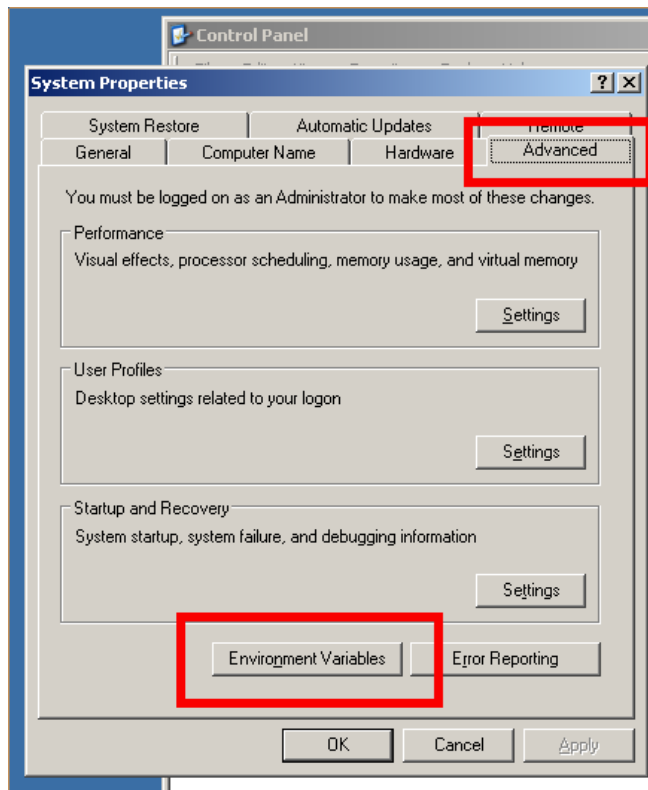


Figure 21. Accessing 'Environment Variables' in Windows XP

4. Select the 'Path' variable then click the 'Edit' button in the 'Environment Variables' dialog box (figure 22).

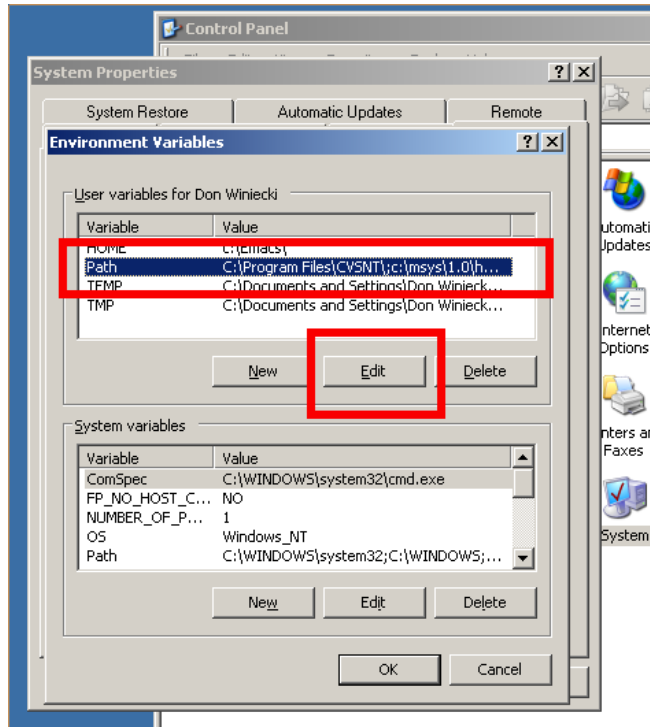


Figure 22. Preparing to edit your Path Environment Variable.

5. In the 'Edit User Variable' dialog box (figure 23), add the following text.

NOTE: The Variable Value should go at the *end* of the Path string.
Semi-colons are required.

Variable Name: Path

Variable Value: ;c:\MinGW\bin;c:\mingw\msys\1.0\bin;

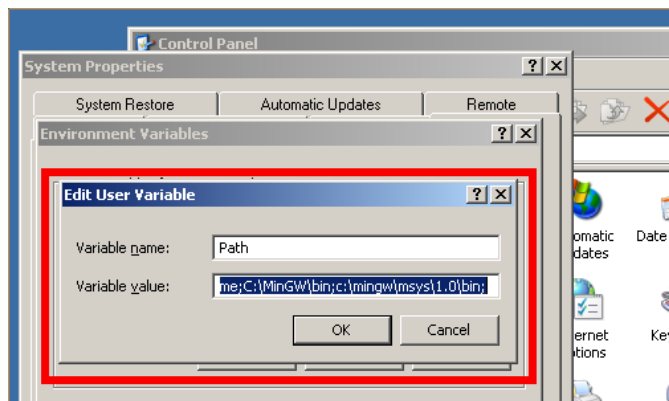


Figure 23. Adding required PATH details in the 'Edit User Variable' dialog box.

6. Close each of the opened dialog boxes as indicated in table 2 below.

Table 2. Closing Dialog Boxes After Setting WinXP Path

Dialog Box Title	Close By Doing This
Edit User Variable (figure 21)	Click the `OK` button
Environment Variables (figure 20)	Click the `OK` button
System Properties (figure 19)	Click the `OK` button
Control Panel (figure 18)	Click the Close Box (the `X` button in the window's title bar)

- 2.17 Verify your MinGW installation by double-clicking the MSYS icon on your desktop (step 2.15, above) and entering the following command into the MinGW shell.

```
gcc -v
```

The last line in output produced by the command above should be `gcc version 4.8.1 (GCC)`.

NOTE: If this is not the case, remove the directory into which you installed MinGW (see step 2.4 above and the note immediately preceding step 2.4 above) and repeat all components of step 2 above, taking care to verify the process.

**You are now finished installing tools for building GCL.
Directions for acquiring source for GCL and building it are included below.**

3. CLONE CURRENT GCL USING Git

- 3.1 Choose or create a directory into which you will download/clone the most current GCL code (I created `c:_git`).
- 3.2 Launch the Git Bash shell by double-clicking its icon on your desktop (figure 4, above).
- 3.3 Change to the directory you created in step 3.1 above (for example, “`cd c:/_git`”)
- 3.4 Enter the following command to initiate the clone process.

```
git clone git://git.sv.gnu.org/gcl.git
```

NOTE: You will see the clone process proceed in the MinGW Git shell (figure 24). When the `\$` prompt reappears in the MinGW Git shell, the process is complete and a GCL code has been saved in a directory named “gcl” in the directory specified in steps 3.1 through 3.3 above.

```

MINGW32:/c/_git
dwiniiecki@DWINIECKI-WIN7 /c/_git
$ git clone git://git.sv.gnu.org/gcl.git
Cloning into 'gcl'...
remote: Counting objects: 59538, done.
remote: Compressing objects: 100% (13146/13146), done.
remote: Total 59538 (delta 39763), reused 59524 (delta 39757)
Receiving objects: 100% (59538/59538), 57.21 MiB | 7.35 MiB/s, done.
Resolving deltas: 100% (39763/39763), done.
Checking connectivity... done
Checking out files: 100% (10223/10223), done.

dwiniiecki@DWINIECKI-WIN7 /c/_git
$ _

```

Figure 24. The Git clone process

4. SELECT/CHECKOUT GCL VERSION TO BUILD

- 4.1 While still in the Git Bash shell, change to the `gcl` directory with the following command:

```
cd gcl
```

- 4.2 Remove system files not needed for Win32 build processes with the following command:

```
rm gcl/debian/control*
```

- 4.3 Checkout the version of GCL you will be building with a command formatted as follows (this command is case-sensitive and must contain underscore characters as shown):

```
git checkout Version_2_6_10
```

NOTE: You will see the checkout process proceed in the MinGW Git shell (figure 25, below). When the '\$' prompt reappears in the shell you are ready to build the version of GCL checked out in step 4.3 above.

```

MINGW32:/c/_git/gcl
dwiniecki@DWINIECKI-WIN7 /c/_git
$ cd gcl

dwiniecki@DWINIECKI-WIN7 /c/_git/gcl
$ rm gcl/debian/control*

dwiniecki@DWINIECKI-WIN7 /c/_git/gcl
$ git checkout Version_2_6_10
Checking out files: 100% (8175/8175), done.
Branch Version_2_6_10 set up to track remote branch Version_2_6_10 from origin.
Switched to a new branch 'Version_2_6_10'

dwiniecki@DWINIECKI-WIN7 /c/_git/gcl (Version_2_6_10)
$

```

Figure 25. Checking out a version of GCL from cloned files

5. BUILDING GCL USING MSYS AS THE HOST

NOTE: The following assumes you have successfully completed steps 1 through 4 above.

- 5.1 Launch the MSYS shell using the icon created in step 2.15 above.
- 5.2 At the MSYS '\$' prompt, change to the 'gcl' directory in which you cloned GCL source code (see steps 3.1 – 3.3 above), for example:

```
cd c:/_git/gcl/gcl
```

You are now ready to build GCL

NOTE: Speed of the process indicated in step 5.3a or 5.3b (below) will vary with the power of your computer. It takes about 12 minutes on my 2.2GHz dual core machine.

- 5.3a If you want to build the CLtL1 variant, enter the following command:

```
./configure && make && make install
```

- 5.3b If you want to build the ANSI variant, enter the following command:

```
./configure --enable-ansi && make && make install
```

- 5.4 When the '\$' prompt reappears in the MinGW shell, verify the process was successful by entering one of the following commands at the MSYS '\$' prompt, depending on the version of GCL you have just attempted to build:

For the CLtL1 version, type the following:

```
./unixport/saved_gcl.exe
```

For the ANSI version, type the following:

```
./unixport/saved_ansi_gcl.exe
```

The result of these commands should be as shown in figure 26.

NOTE: If after completing the above you see any other output than shown in figure 26, please perform steps 5.5 through 5.9 to produce a log of the build process that can be sent to the GCL maintainers for diagnosis. Otherwise, read the NOTES indicated below and proceed to step 6.

NOTE: After confirming that the build process has completed successfully, type (bye) or (by) at the GCL prompt to close GCL.

NOTE: BFD fasloading, Stratified Garbage Collection (SGC) readline and GCL-TK do not work under Win32. The build process indicated here provides a GCL executable which will build current versions of available CAS systems.

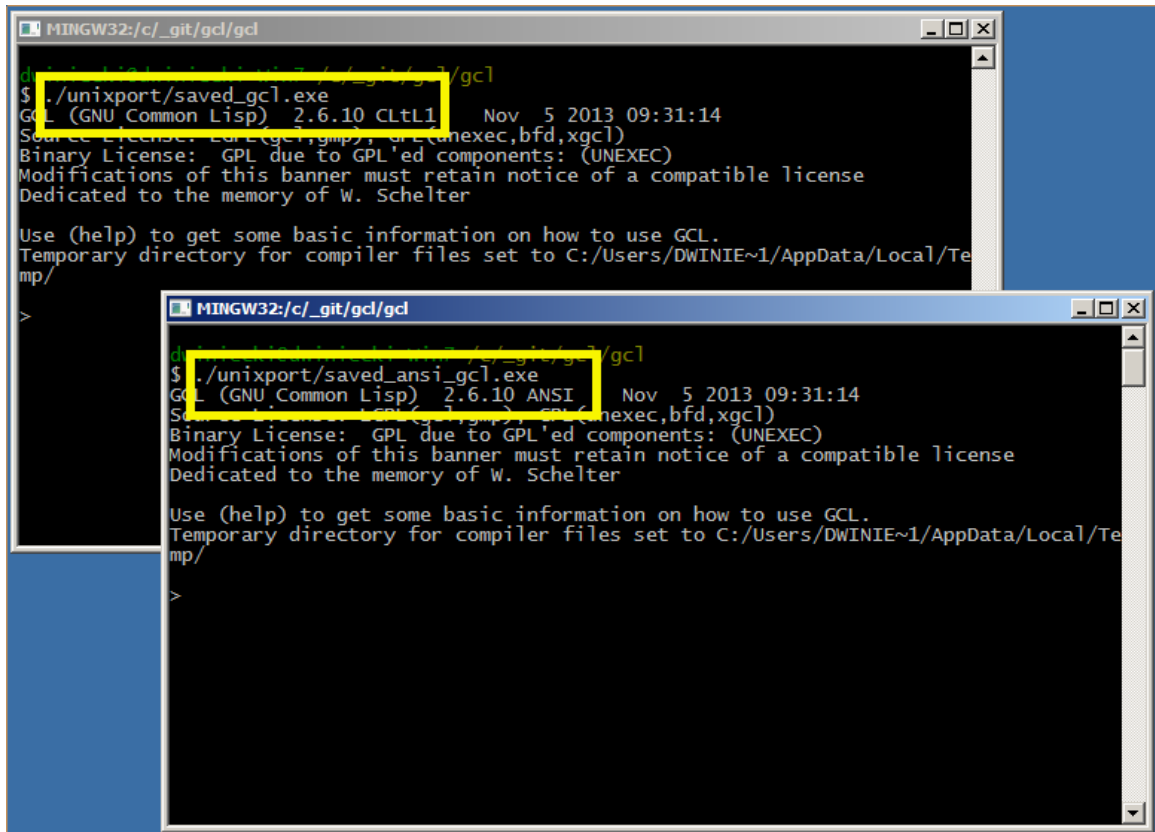


Figure 26. When successfully compiled, GCL will display this on start.

5.5 At the MSYS '\$' prompt, type the following:

```
make clean
```

5.6 When the MSYS '\$' reappears, type the following command:

NOTE: You will not see visible output in the MSYS shell for steps 5.6 through 5.8. Output is redirected to the files named in each command.

```
./configure --enable-ansi &>c:/config.log
```

5.7 When the MSYS '\$' reappears, type the following command:

```
make &>c:/make.log
```

5.8 When the MSYS '\$' reappears, type the following command:

```
make install &>c:/install.log
```

5.9 During steps 5.6 through 5.8, files named `config.log`, `make.log`, `install.log` will be saved at the `c:\` (root) level of your computer's hard drive.

Copy these files to an E-mail message and send them to the GCL-devel listserv at the following E-mail address.

```
gcl-devel@gnu.org
```

You should also include the following information with your E-mail:

- The version of GCL you are attempting to build
- The operating system of the computer on which you are attempting to build GCL
- The software you will build with GCL (for example: ACL2, Axiom, Open-Axiom, Maxima, etc.)

6. INSTALLING GCL INFORMATION ACCESSIBLE FROM GCL INTERPRETER

6.1 Download the file named `gcl.info.tgz` from `ftp://ftp.gnu.org/pub/gnu/gcl/`

6.2 Make the following directory path and unpack contents of the above-named file into it:

```
c:\usr\local\gnu\info
```

7. RUNNING GCL

You can move the directory containing executable files anywhere on your hard drive.

After moving the directory containing executable files to your desired location, create a shortcut to the executable (either `saved_gcl.exe` or `saved_ansi_gcl.exe`) and assign to it the icon located in the `...\gcl\gcl\bin` directory.

You can then move the shortcut anywhere (for example, the 'Start' menu or your 'Quick Launch' menu). Invoke the icon to launch GCL.

8. CONFIGURE GCL TO RUN IN INFERIOR MODE UNDER EMACS

Follow directions provided by Gordon Novak Jr. at the following URL:

`http://www.cs.utexas.edu/users/novak/gclwin.html`