The git And gcc Commands

Learn C With Babbo

Now that we've become loosely acquainted with Bash, we need to install two commands for it that will be essential this course. These commands are `git` and `gcc`. Refer to your operating system's subsection for how to install them. Note that there are no sections provided that describe how to install `git` and `gcc` on Windows systems that cannot install the Ubuntu subsystem, or Linux machines which are not Ubuntu, but there are plenty of resources available online for those interested. Note that it is likely if you are running Linux that you already have `git` and `gcc` installed.

Windows and Ubuntu

To install `git` and `gcc` on Windows running the Ubuntu subsystem or an Ubuntu machine, enter the following lines in Bash, which will prompt you for your UNIX user password:

```
sudo apt-get update
sudo apt-get install git
sudo apt-get install gcc
```

After you've run these lines, enter `git` and you should see some text describing how to use `git`. If instead you see something similar to "command not found", something went wrong with your installation.
Likewise, enter `gcc`, and you should see something like "fatal error: no input files" if the installation succeeded. If it says command not found or something similar, `gcc` was not properly installed.

You won't need to know `sudo` or `apt-get` to learn C, but I'll give some brief context for those interested. The command `sudo` means "do as superuser" (su = superuser, do = do). It is basically synonymous with something like "Run as Adminstrator" that you may have seen on Windows: whatever follows `sudo` is executed as a privileged/super user named "root", and running as root allows you to perform tasks that require special permissions, such as those for administrators, like installing software.

The command `apt-get` is a software management utility for Ubuntu (not Bash in general) that lets you install software directly from Bash. The first command `sudo apt-get update` updates the list of available software packages, so that if new software became available since the last time you ran `update`, you will be able to install it. The next two commands are self-explanatory: they utilize `apt-get install` to install `git` and `gcc`.

Mac

`git` and `gcc` are provided on a Mac system through the Xcode development tools. To install them, go to the APP Store and search for Xcode and install it. After installation is complete, open a new Terminal (do not use a Terminal that was already open) and enter `git`. You should see text describing how to use `git`. Similarly, enter `gcc`, after which you should see something similar to "fatal: no input files." If either of these commands output something like "command not found", the installation was probably unsuccessful and you may need to troubleshoot online.