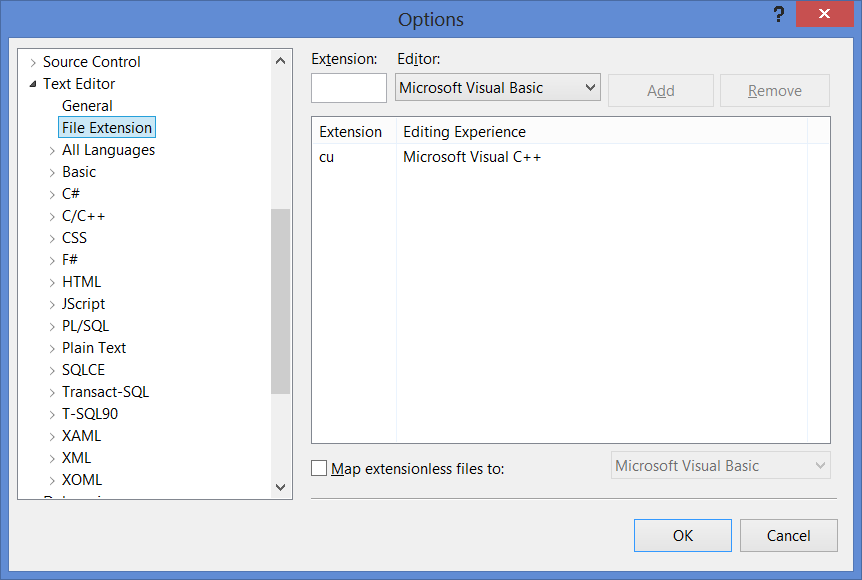
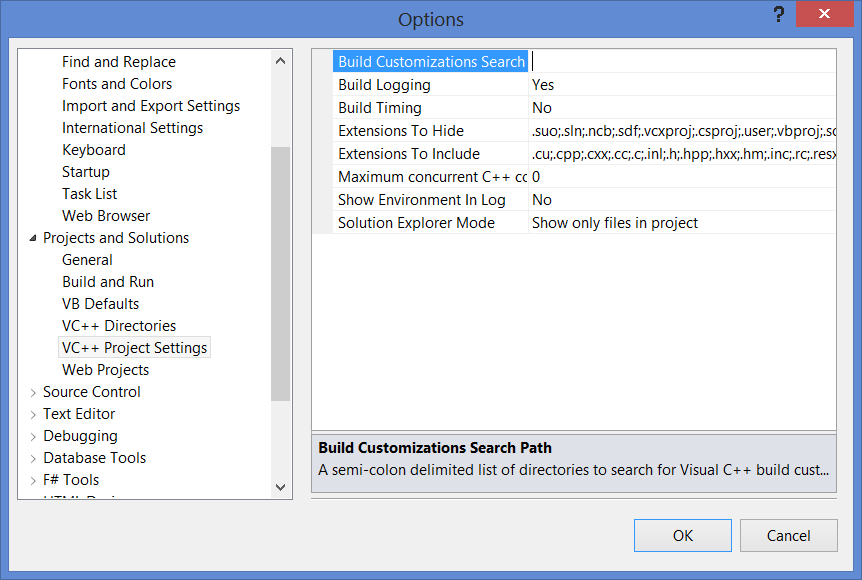
**If you don’t have Visual Studio…**

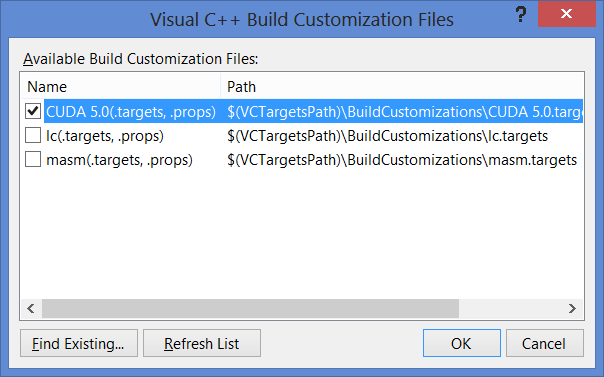
If you wish to read the lab code and you do not have Visual Studio, I suggest using Notepad++ and set the language to C++ to make the code readable with colour coding etc.

**Some Visual Studio 2010 tips for CUDA**

After installing the CUDA Toolkit, you should make the following settings changes in Visual Studio..

* Open Visual Studio 2010 as an Administrator
* Tools / Options / Text Editor / File Extensions , add “cu” as a C++ type :
* 
* Tools / Projects and Solutions / VC++ Project Settings , add “.cu” to extensions to include:
* 
* Close and Restart visual Studio.
* This will colour-code your “.cu” program similar to C++, and it will get a degree of Intellisense working.

Make sure your new CUDA project is of type NVIDIA / CUDA , not a normal C++ project! If you right click the project and view it’s Build Customizations, it should be set to CUDA 5.0 (or whatever toolkit is installed). This ensures the correct compiler switches and libraries are linked for CUDA programming.



CUDA program files are C/C++, but they MUST have the extension “.cu” or they will not be recognised by NVIDIAs special nvcc compiler.

Renaming a “.cpp” file to a “.cu” within a project does not work. Instead, you need to detach the cpp file from the project (do not delete it from disk), rename it in Windows explorer, and then reattach it to your project in Visual Studio using (project/add existing item).

The best way to get our lab examples working is to start a new Visual Studio NVIDIA CUDA project, and then exclude the default “kernel.cu” from the project. Using Windows Explorer, copy the lab files into the project folder. In Visual Studio, add “existing items” to your project, browse to the files and add them.

Be aware that if you add “existing items” to a C/C++/CUDA project in visual studio, it does NOT copy the files into your project folder, as you might expect. Instead it maintains a link to the files in their original location. This can cause confusion when you browse to a project folder expecting to see the files there, and find they are not! That’s why it is best to manually copy files into the project folder before adding them to the project.

Make sure you are not trying to compile files on a network location! The project directory must be local.