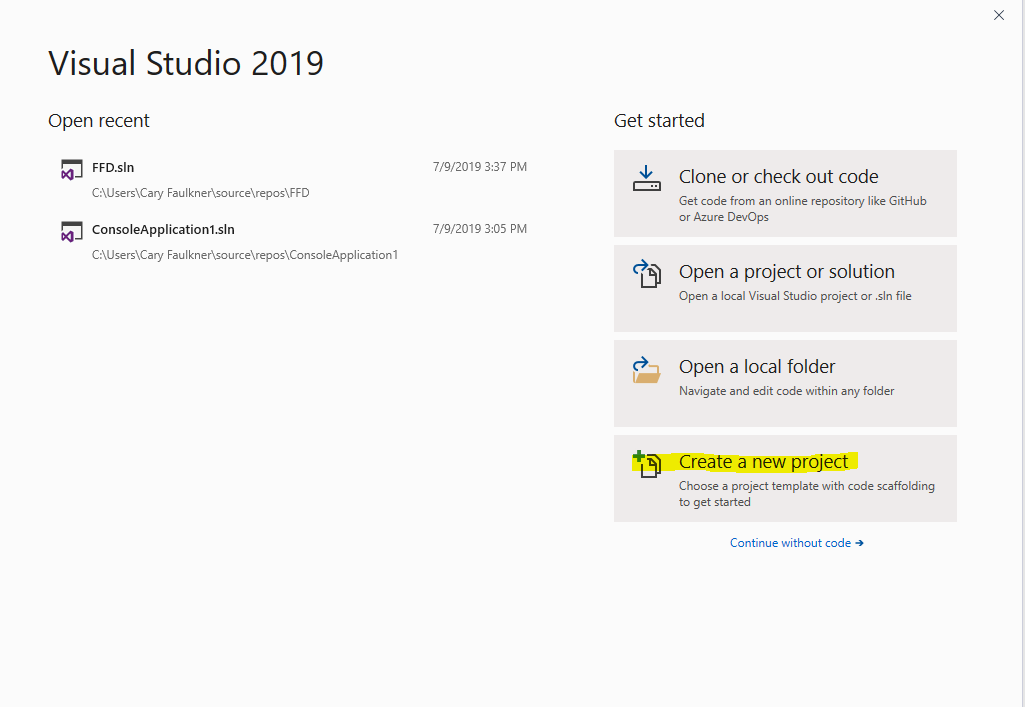
**Guide to Compiling FFD Code Using Visual Studio**

*This guide covers:*

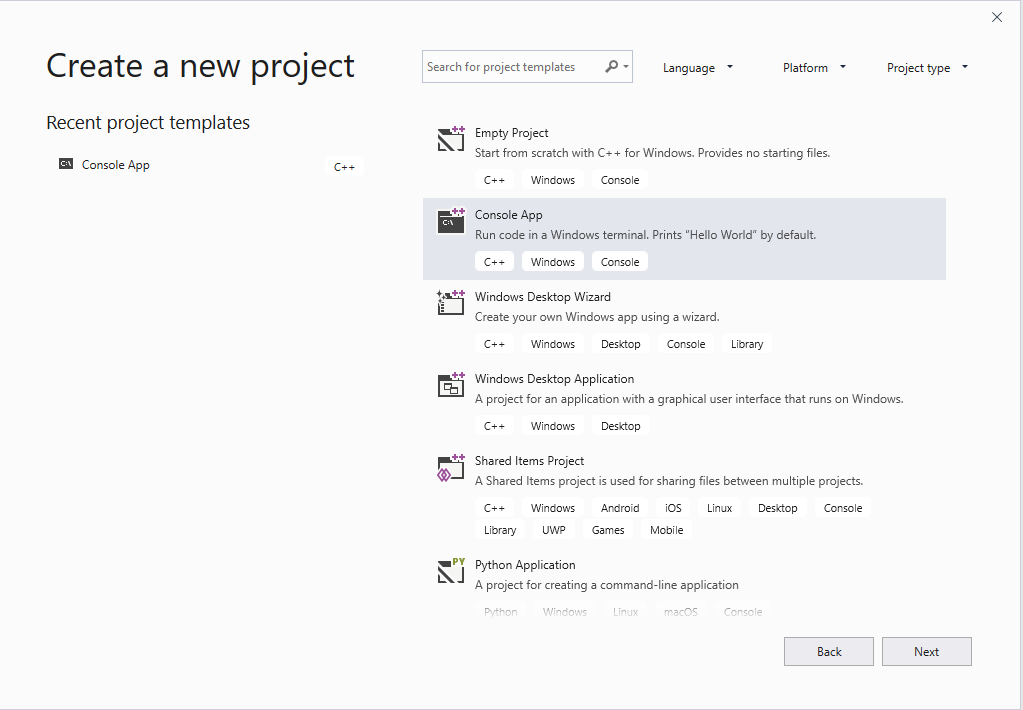
1. *Compiling the CPU code in Visual Studio.*
2. *Compiling the GPU code in Visual Studio.*

*Compiling the CPU code*

1. Begin by opening Visual Studio and selecting “Create a new project.”

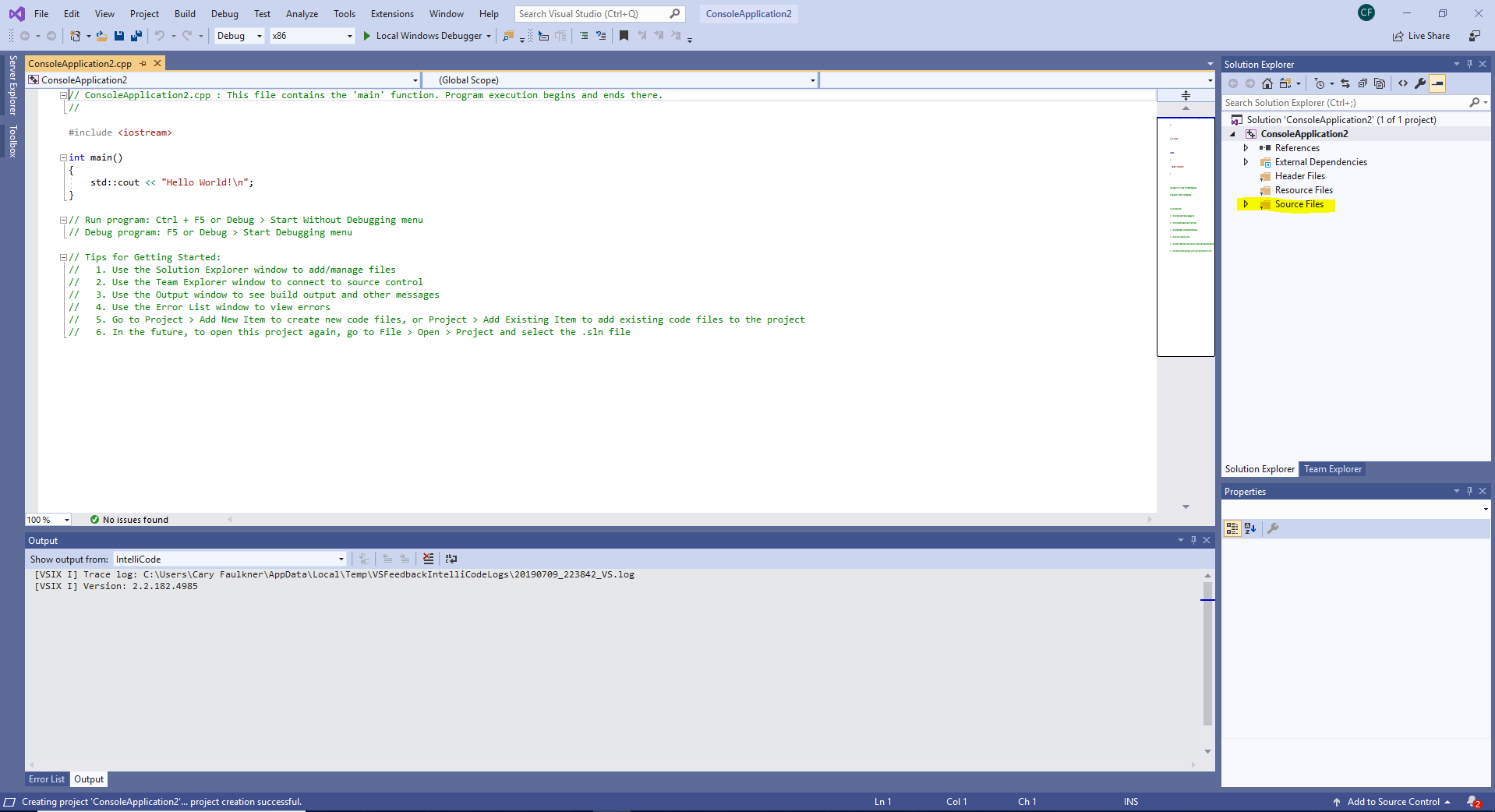
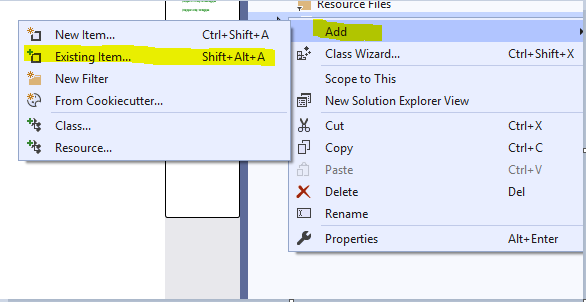


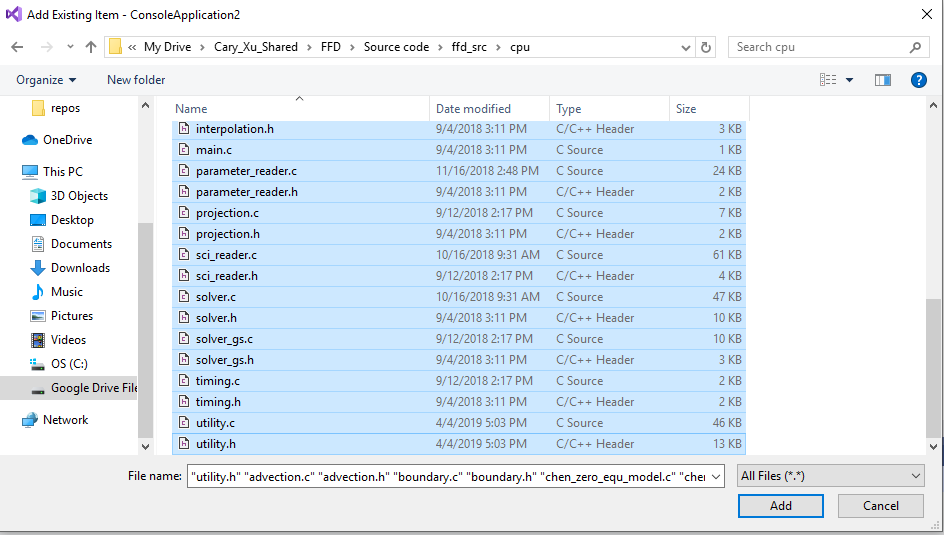
1. Select “Console App” and click “Next.” **NOTE: If Console App is not installed see Appendix A for instructions on how to install it.**

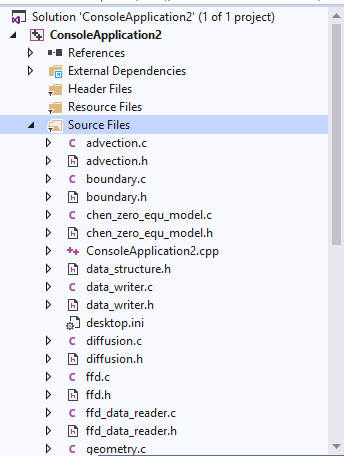


1. Name your project and select its location.

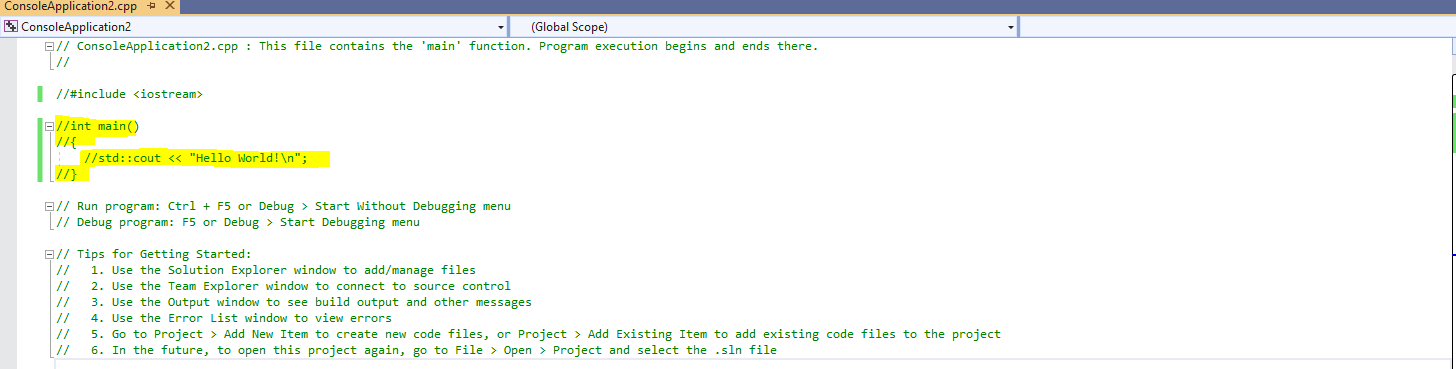


1. The project will open, and we will now add the FFD CPU source code. Right-click on “Source Files” on the right side of the screen. Go to “Add” and select “Existing Item…” 
2. Make sure to add all the FFD source code files for using FFD on CPU. You will see all the files appear on the right side under “Source Files.”

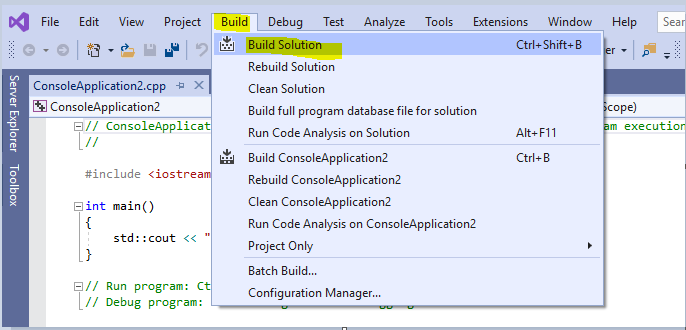




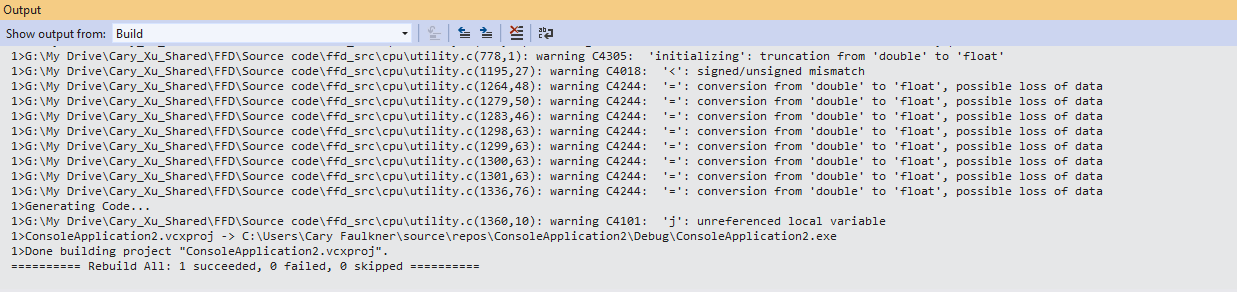
1. It is also important to delete or comment out the default “main” function that was created when you first opened the project, as this may cause errors when compiling the code. Use “//” to comment out code.



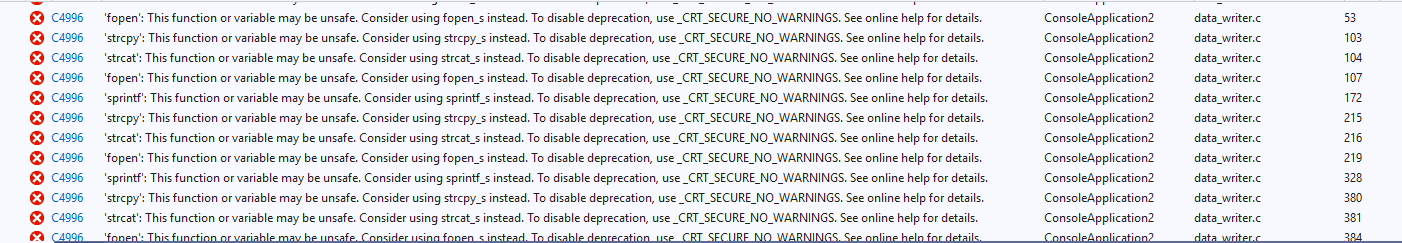
1. To compile the code, select “Build” from the menu in the top left, and click “Build Solution.”



1. The code should, hopefully, compile with no errors and the output should look as shown below.

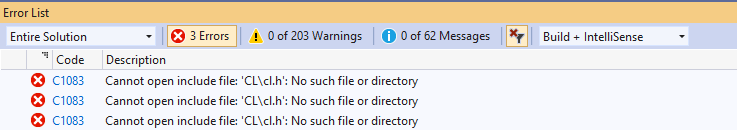


1. You may receive many errors with the same description, as shown below. See Appendix B on how to fix this.

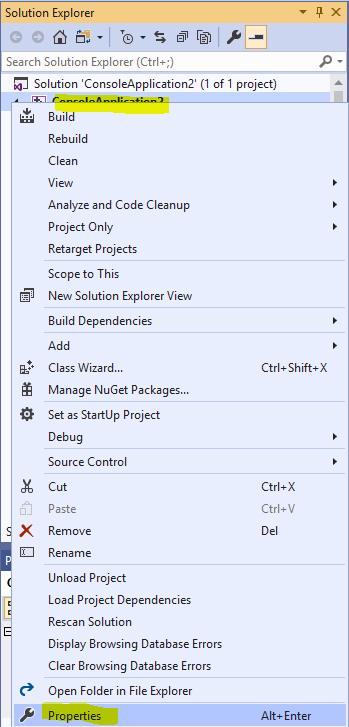


*Compiling the GPU Code*

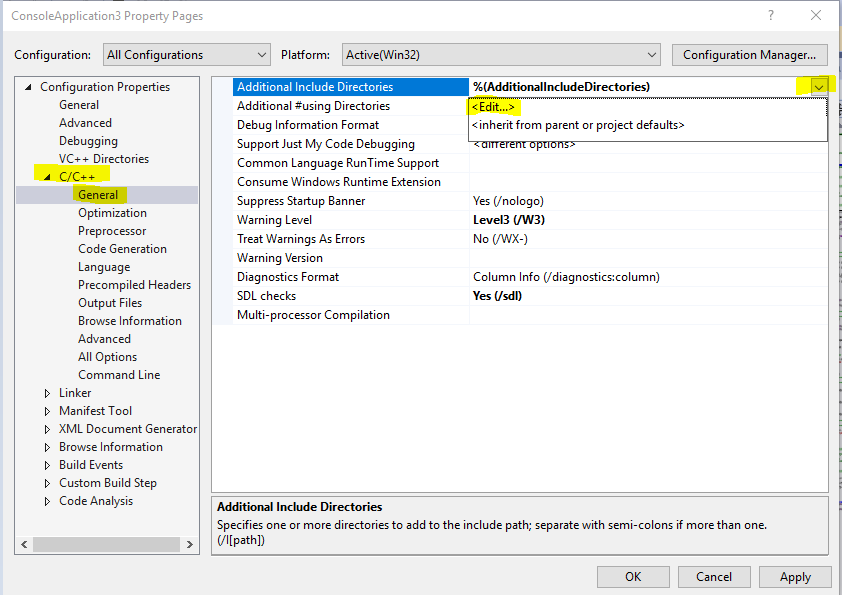
1. To begin, repeat steps 1-6 from compiling the CPU code, except in step 5 make sure to add the source files for using FFD on GPU rather than CPU. If the following files are added, remove them as they will cause errors: opencl\_iso\_main.c, main.c, and platform\_device\_detect.c. **Do NOT remove opencl\_main.c.**
2. Try compiling the code as shown in step 7 of the CPU section. If you get the same set of errors as shown in step 9 of the CPU section, follow the directions in Appendix B and recompile the code. After doing this, you will likely get the error shown below:



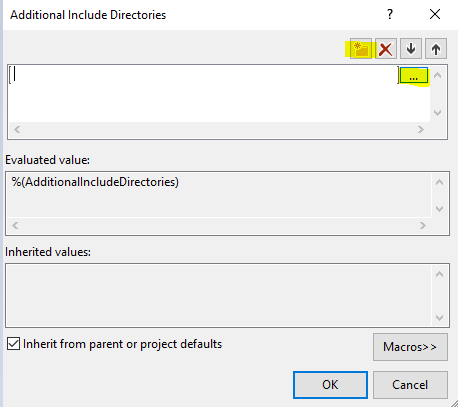
1. To fix this error we need to first install OpenCL. The version of OpenCL you install is based on your GPU. My GPU is AMD Radeon, so I installed OCL\_SDK\_Light\_AMD.exe from: <https://github.com/GPUOpen-LibrariesAndSDKs/OCL-SDK/releases>. If your GPU is NVIDIA, you can find it here: <https://developer.nvidia.com/opencl>. Otherwise, simply Google search your GPU + OpenCL to find the right version of OpenCL for your GPU.
2. We now need to add the path of OpenCL. Right-click on the name of your project on the right side and select “Properties.”

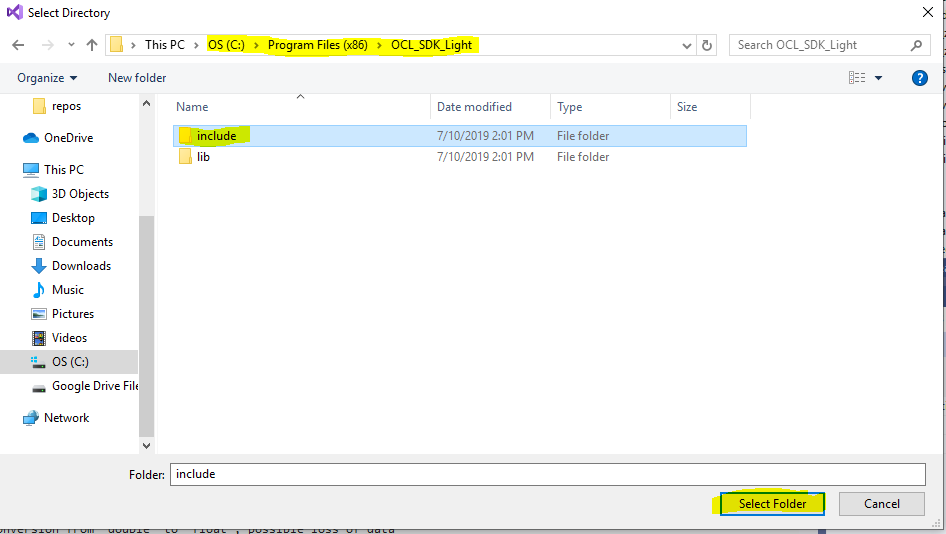


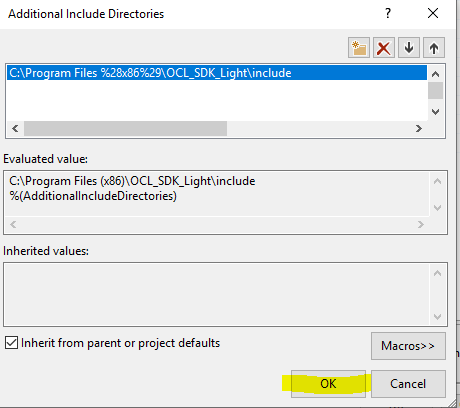
1. Under C/C++, select “General” and click the drop-down menu for “Additional Include Directories” and select <Edit…>



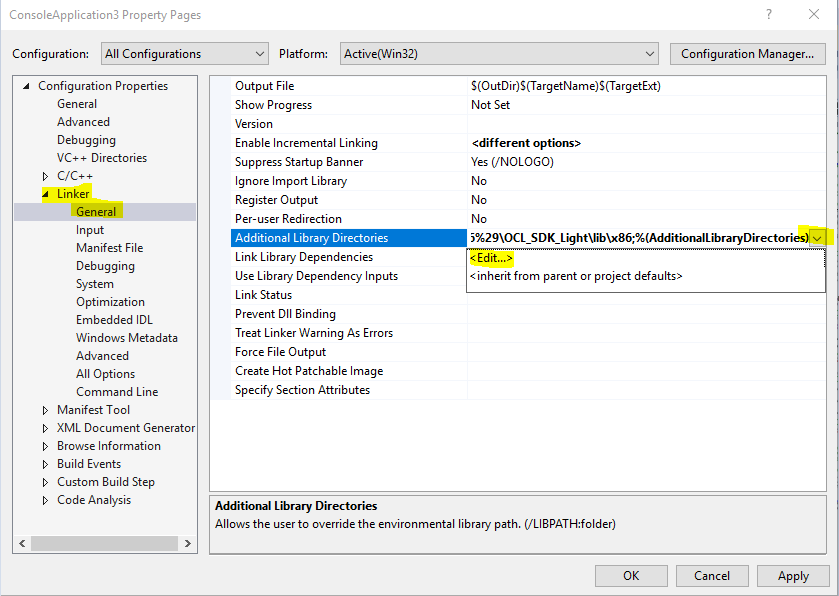
1. Click the folder icon and the three dots to add a new path. We now need to select the path for our CL folder which contains cl.h. For me this is in C:\Program Files (x86)\OCL\_SDK\_Light\include. Once you have located the folder, click “Select Folder” and then “OK.”

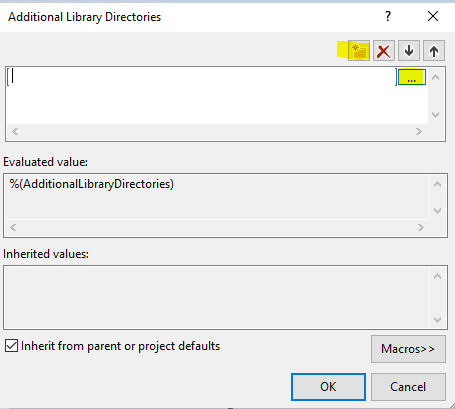


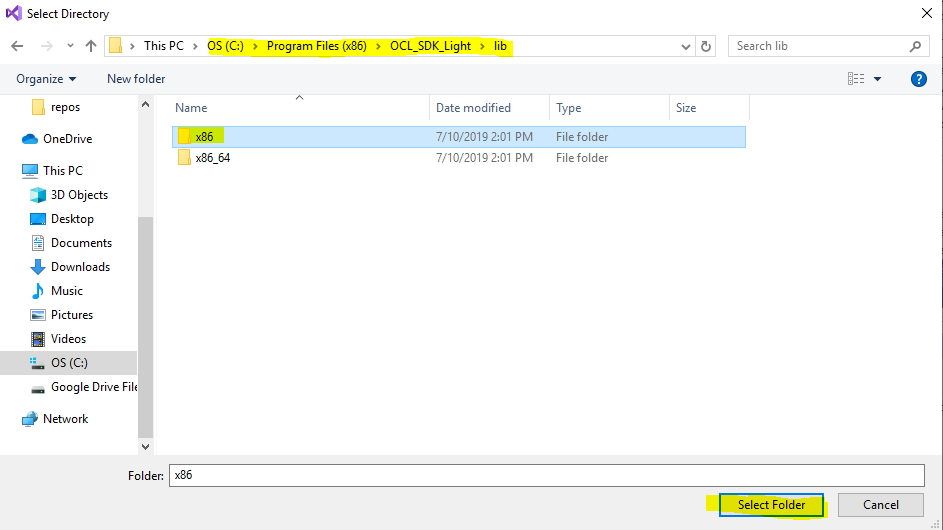


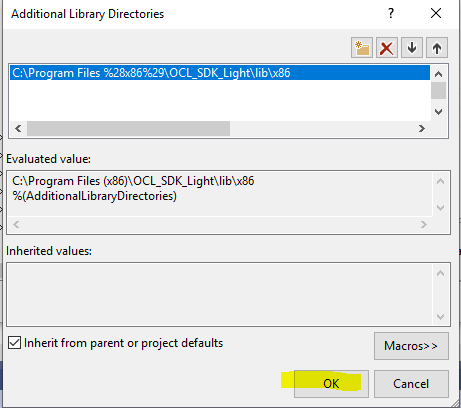


1. Under “Linker”, select “General” and click the drop-down menu for “Additional Library Directories” and click <Edit…>. Once again, click the folder icon and the three dots. This time we will select the path for the opencl.lib file. For me this is in C:\Program Files (x86)\ OCL\_SDK\_Light\lib\x86. Click “Select Folder” then “OK.”

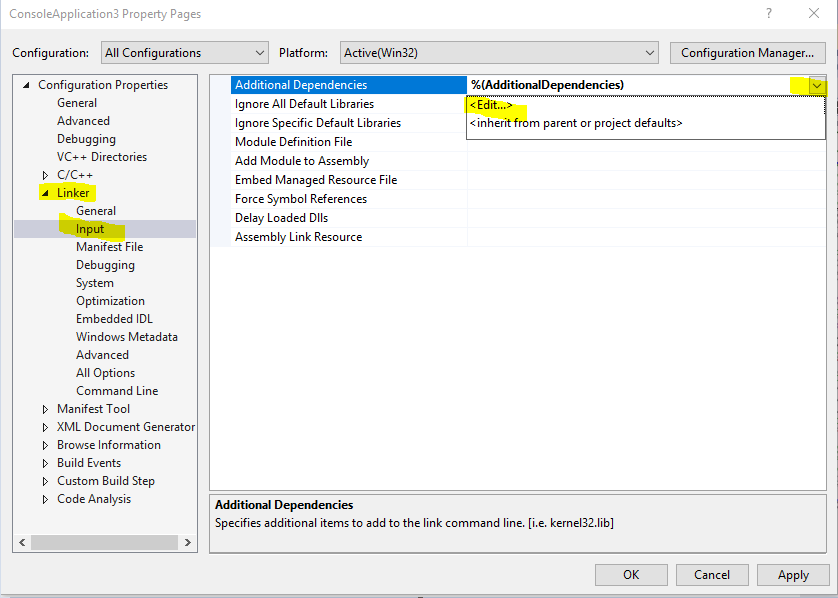


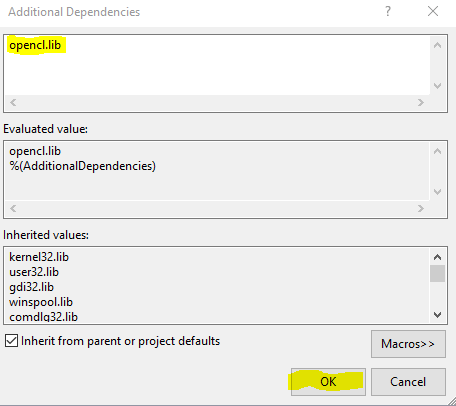






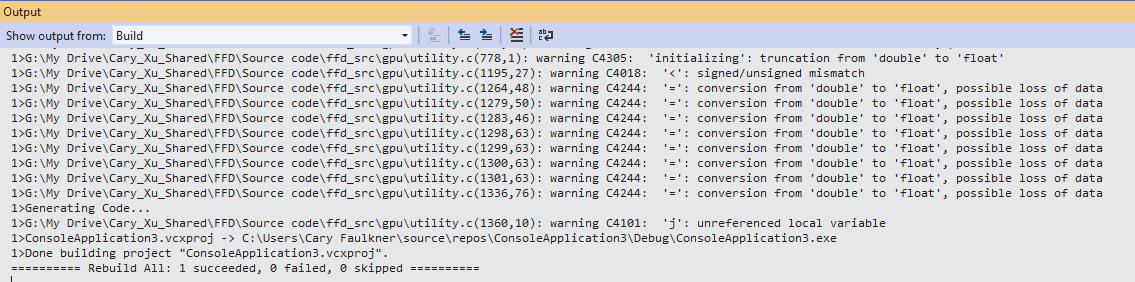
1. Under Linker again, select “Input” and click the drop-down menu for “Additional Dependencies” and click <Edit…>. Type in opencl.lib and click “OK.”





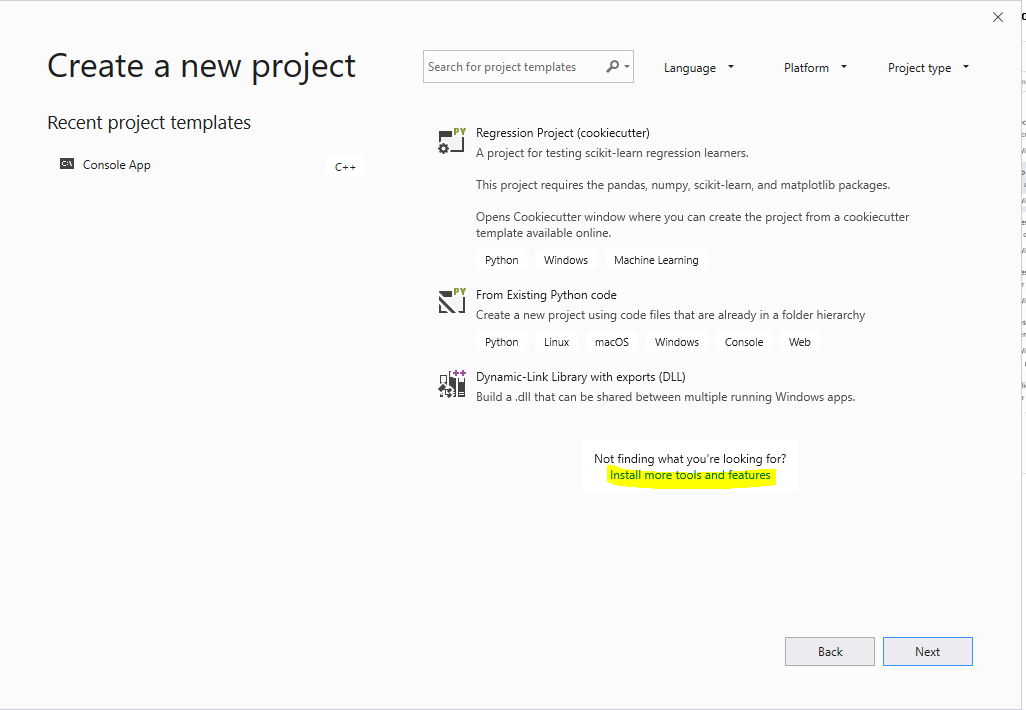
1. Click “Apply” then “OK”. When you recompile the code there should be no errors.



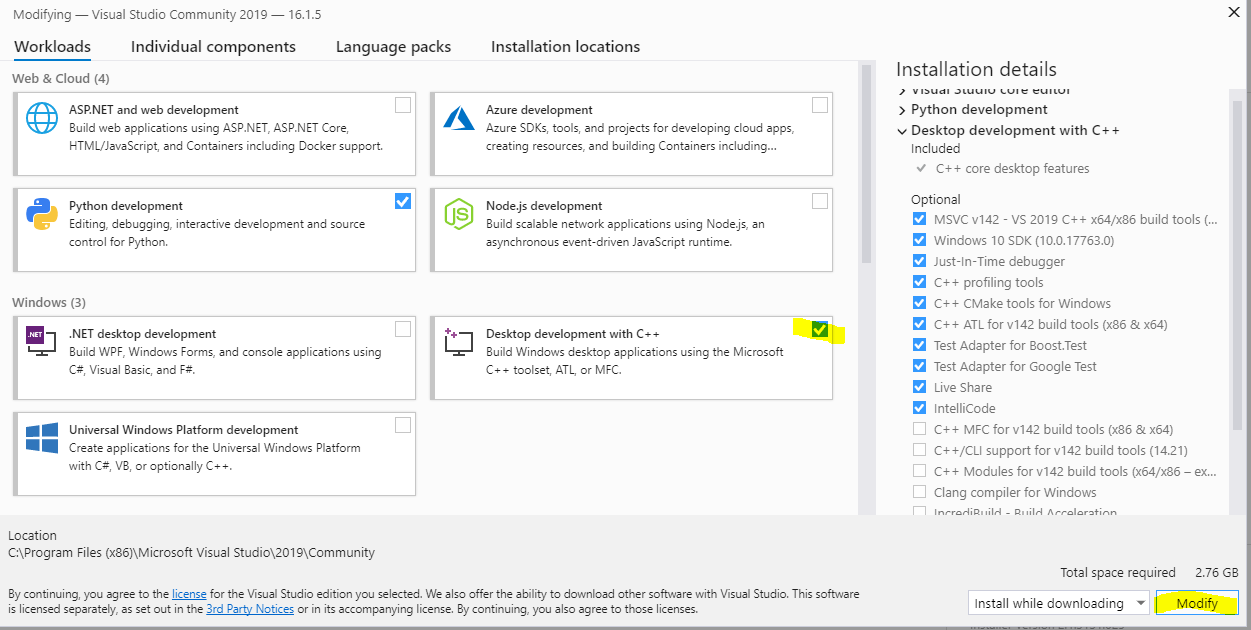


**Appendix A: Installing the Console App**

1. If the Console App is not installed, scroll down and select “Install more tools and features”

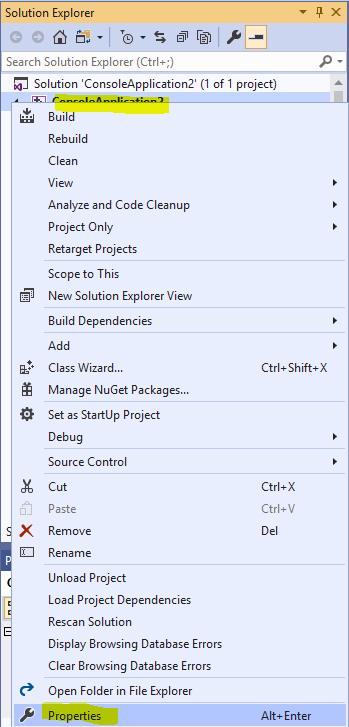


1. Check “Desktop development with C++” and click “Modify” in the bottom right corner. The package will install, and it may ask you to restart your computer. Once completed, return to step 2 of the main section.

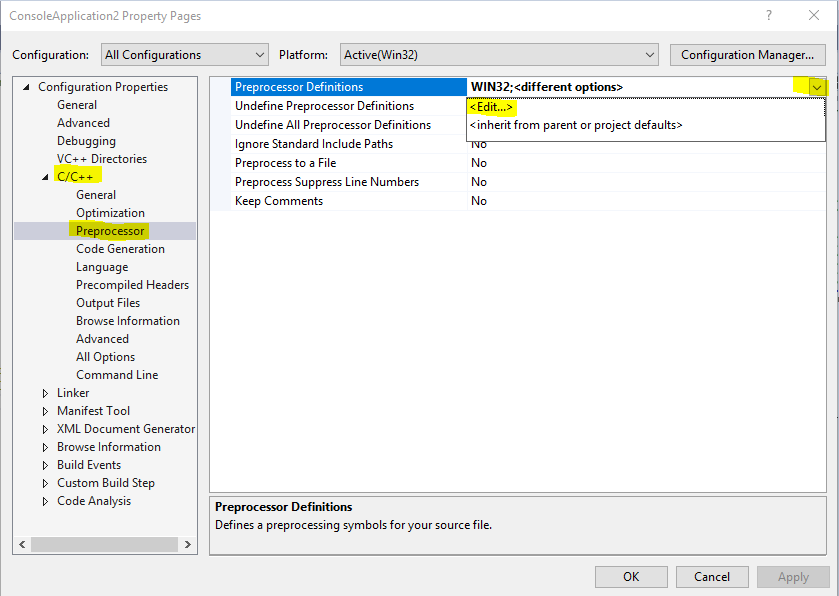


**Appendix B: Disabling “\_CRT\_SECURE\_NO\_WARNINGS” Errors**

1. To disable these errors, right-click on the name of your project on the right side and select “Properties”



1. Select “Preprocessor” on the left side under C/C++, select the drop-down menu for “Preprocessor Definitions” and click <Edit…>.



1. Type in \_CRT\_SECURE\_NO\_WARNINGS as shown and click “OK.” On the properties page, select “Apply” then “OK.” When you rebuild the solution now (recompile the code), these errors will not appear.

