# 600086 Lab Book

# Week 2 – CUDA Lab 4. CUDA OpenGL Interoperability & Image processing

Date: 24th Feb 2022

## Exercise 1. Create an OpenGL-CUDA program based on a CUDA SDK sample

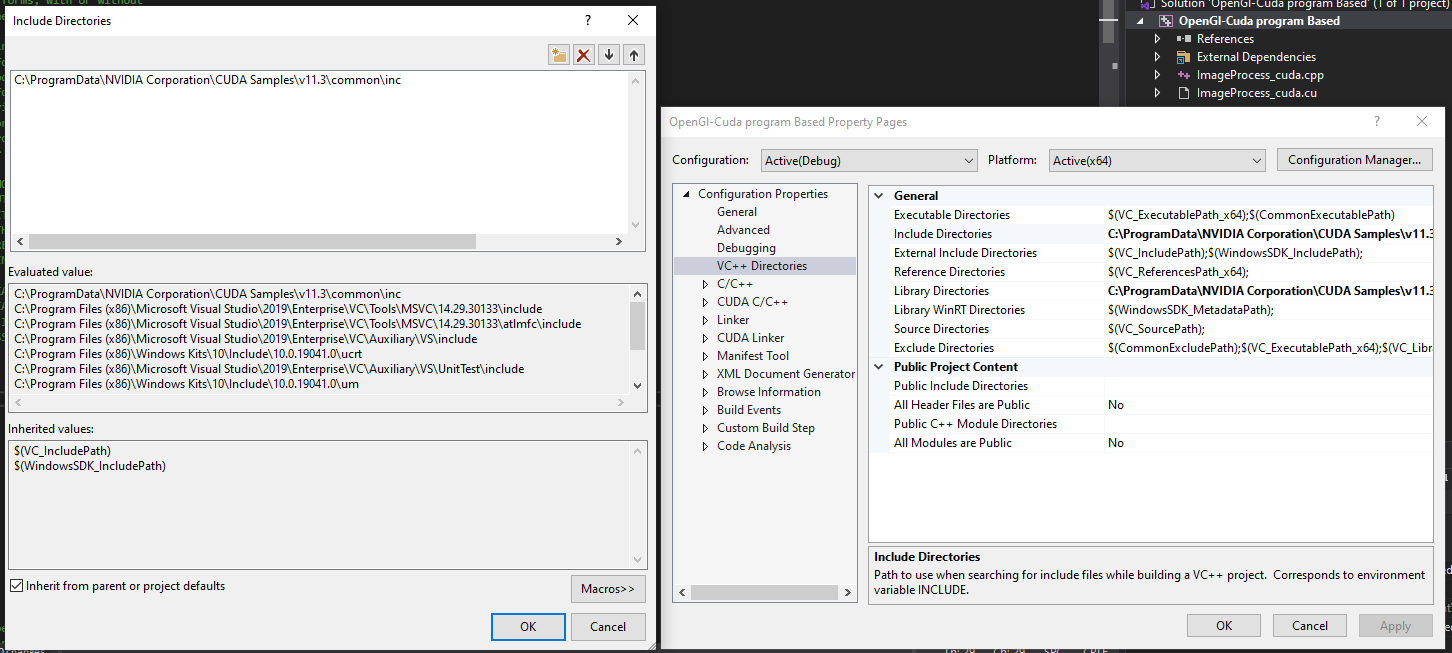
### Question:

Create an OpenGL-CUDA program based on a CUDA SDK sample

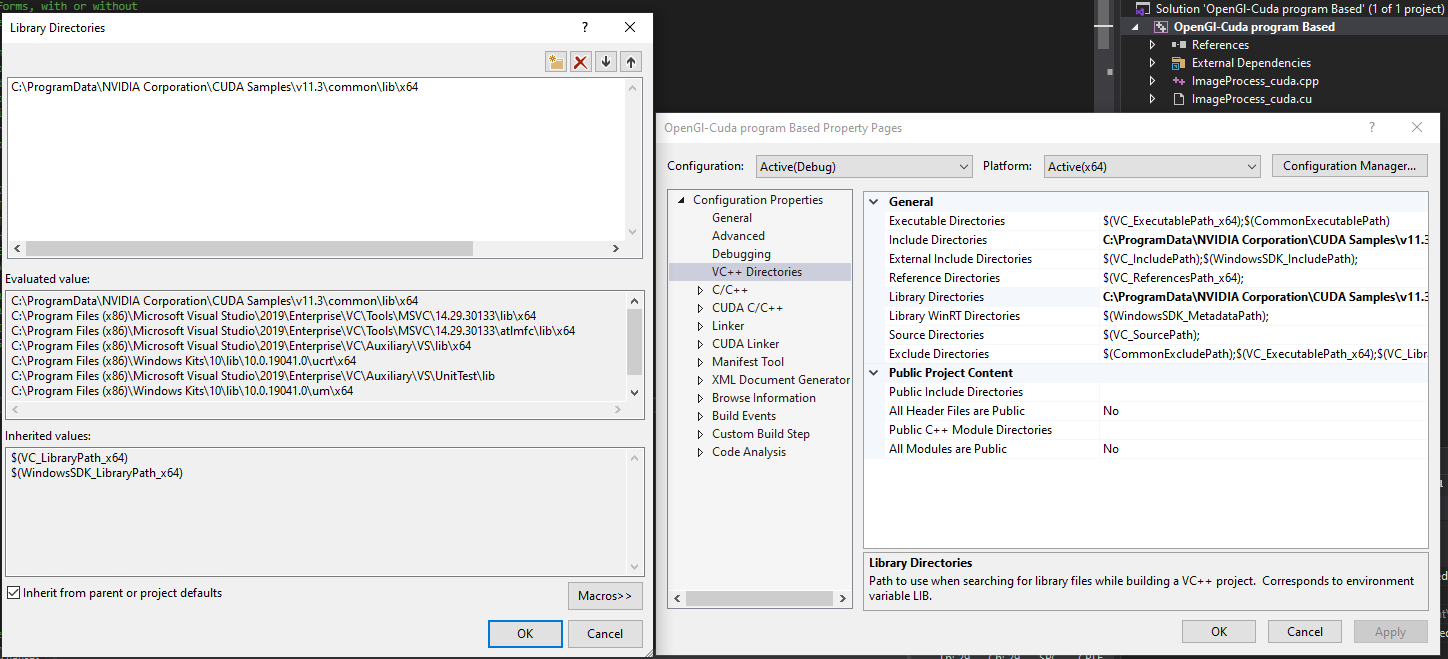
### Solution:

No sample code to show

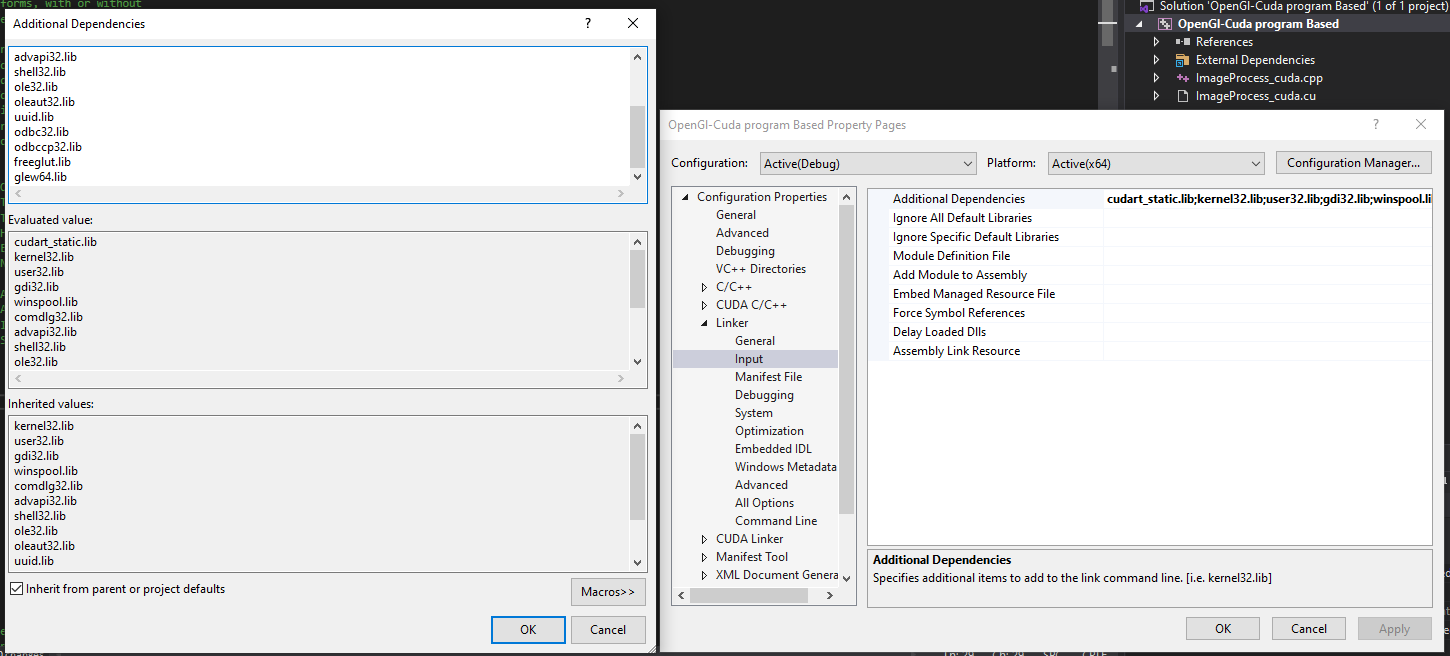
Step 1 : adding include directories to project



Step 2 : adding Lib directories



Step 3 : adding to the Linker files



Step 4 : I then compiled the project resulting in the command line ouput shown in sample output data

### Test data:

n/a

### Sample output:

### 

### Reflection:

Nothing to report was fairly perfunctory

### Metadata:

N/A

### Further information:

N/A

## Exercise 2. Understand pixel colour

## Question:

a) An image is simply a 2D array of pixels. Each pixel has a colour value which can be digitally

represented as a list of numbers, depending on the data format adopted. In the framework, the

Colour of each pixel is represented in RGBA format using 4 integers, each of which ranging from 0

to 255. Open ImageProcess\_cuda.cu and go to the method d\_render( ), modify the 4 numbers

shown in make\_uchar4( ..., ..., ..., ... ) in the following line:

d\_output[i] = make\_uchar4(c \* 0xff, c \* 0xff, c \* 0xff, 0);

say,

d\_output[i] = make\_uchar4(0xff, 0, 0, 0);   
 and then

d\_output[i] = make\_uchar4(0, 0xff, 0, 0);   
d\_output[i] = make\_uchar4(0, 0, 0xff, 0);

b) The original image is a grey value image, the pixel intensity at a pixel position at (u,v) is read   
 using   
 float c = tex2DFastBicubic<uchar, float>(texObj, u, v);

where c is in [0, 1].

c) Now modify the value d\_output[i] using image pixel value c read from image location at

(u, v) with the following colour and observe how the image colour is changed.

d\_output[i] = make\_uchar4(0, 0, c\*0xff, 0);