GPU and GPGPU Programming

Shuai Lu 170742

Assignment 1

Task
1. Setup
   Install Visual Studio 2015 or higher or choose other suitable IDE for C++.
   Install CUDA from https://developer.nvidia.com/cuda-downloads
   Install CMake.
   From the folder of the assignment (where the CMakeLists.txt file is located) run the command 'cmake .' to generate a project.
   Compile and run
2. OpenGL Query and print (to console):
   OpenGL version and available extensions:
   GL vendor, renderer, and version
   Find out how to query extensions with GLEW (http://glew.sourceforge.net/, http://www.opengl.org/registry/).
   Query and print the extensions your GPU supports.
   Query and print GPU OpenGL limits:
      maximum number of vertex shader attributes
      maximum number of varying floats
      number of texture image units (in vertex shader and in fragment shader, respectively)
      maximum 2D texture size
      maximum number of draw buffers
      other information of your choice
3. CUDA Query and print CUDA functionality:
   number of CUDA-capable GPUs in your system using cudaGetDeviceCount()
   CUDA device properties for every GPU found using cudaGetDeviceProperties():
   device name
   compute capability: driver version, runtime version, major and minor rev. numbers
   multi-processor count
   clock rate
   total global memory
   shared memory per block
   num registers per block
   warp size (in threads)
   max threads per block
4. GLFW Read up on GLFW. Try to implement the callbacks for mouse interaction and keyboard events. Simply print to the console that you have detected a mouse move/click or a keyboard event.
5. Commit and push your solution and a short report that includes the output of your program

Outcome:
This assignment is finished on Linux environment. The related repositories and software are installed successfully.

The OpenGL vendor, renderer, version and extensions are queried and printed as shown in Fig.1.

GPU OpenGL limits are queried and printed in Fig.2.

CUDA functionality is queried using function cudaGetDeviceProperties(). The information is printed as follow:

The window is created by using GLFW framework. The position of the cursor, the behaviors of the mouse and the keyboard are traced by using function glfwSetCursorPosCallback(), glfwSetMouseButtonCallback(), and glfwSetKeyCallback(). The related callback functions are also created. The detected information is printed on the console(Fig.4).

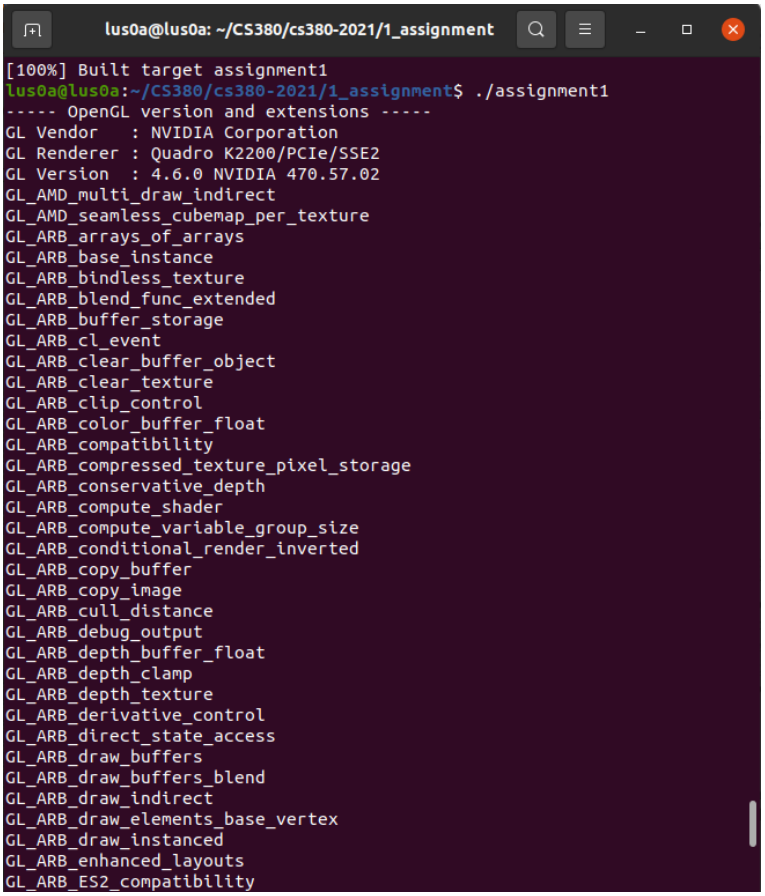

Figure 1: OpenGL query and print

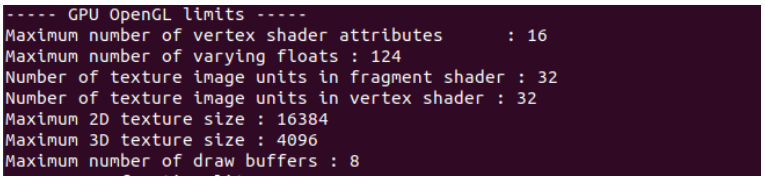

Figure 2: GPU OpenGL limits

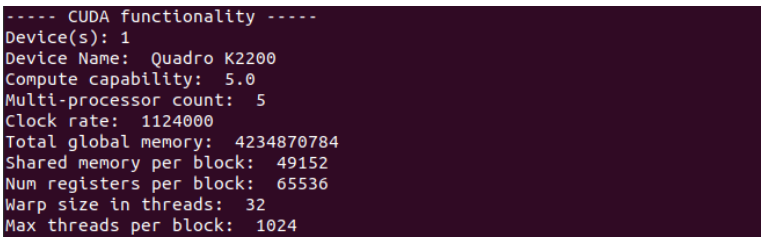

Figure 3: CUDA functionality

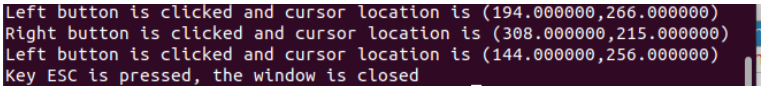

Figure 4: The record of mouse and keyboard events