جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology

Course Syllabus: GPU and GPGPU Programming - CS 380

Division	Computer, Electrical and Mathematical Sciences & Engineering
Course Number	CS 380
Course Title	GPU and GPGPU Programming
Academic Semester	Fall
Academic Year	2019/2020
Semester Start Date	08/25/2019
Semester End Date	12/10/2019
Class Schedule (Days & Time)	10:30 AM - 12:00 PM | Mon Wed

Instructor(s)

Name	Email	Phone	Office Location	Office Hours
Markus Hadwiger	markus.hadwiger@kaust.edu.sa	+966128080260		Come by my office (Bldg 1, Room 2119) or send me an email to make an appointment.

Teaching Assistant(s)

Name	Email
Peter Rautek	peter.rautek@kaust.edu.sa

Course Information

Comprehensive Course Description	This course covers the architecture and programming of GPUs (Graphics Processing Units). It covers both the traditional use for rendering graphics, as well as the use of GPUs for general purpose computations (GPGPU), or GPU Computing. We will cover the basic architecture and the programming model of GPUs. We will cover both the traditional use of GPUs for graphics and visualization, as well as their use for general purpose computations (GPGPU). We will cover GPU many-core hardware architectures, shading and compute programming languages and APIs, programming vertex, geometry, and fragment shaders, programming with CUDA, Brook, OpenCL, stream computing, approaches to massively parallel computations, memory subsystems and caches, rasterization, texture mapping, linear algebra computations, alternative and future architectures. A very important part of the course are the programming exercises, where you will use OpenGL/GLSL and CUDA for a variety of assignments and a semester project.
Course Description from Program Guide	The course covers the architecture and programming of GPUs (GraphicsProcessing Units). It covers both the traditional use of GPUs forgraphics and visualization, as well as their use for general purpose computations (GPGPU, GPU Computing). The main contents are: GPUmany-core hardware architecture, shading and GPU programming languages and APIs, programming vertex, geometry, and fragment shaders, programming with CUDA, Brook, OpenCL, stream computing, approaches to massively parallel computations, memory subsystems and caches,rasterization, texture mapping, linear algebra computations, alternative and future architectures.
Goals and Objectives	The goal is to convey a deep understanding of GPU architecture and APIs (OpenGL, GLSL, CUDA) with important practical applications. The goal is an understanding of both the traditional use of GPUs for rendering graphics, as well as the use of GPUs for general purpose computations (GPGPU), or GPU Computing. By the end of the course you should understand GPU architectures in detail and be able to implement non-trivial GPU programs for graphics as well as for computation.
Required Knowledge	Most important are programming skills, preferentially low-level programming knowledge in C/C++. An understanding of basic computer architecture (processors, memory hierarchies, virtual memory, etc.) is also important.

Reference Texts	www.amazon.com/OpenGL-Shading-Language-Cookbook-Second/dp/1782167021/ www.amazon.com/Programming-Massively-Parallel-Processors-Hands/dp/0128119861/
Method of evaluation	**40.00**% - Homework /Assignments **30.00**% - Quiz(zes) **30.00**% - Course Project(s)
Nature of the assignments	There will be 4 programming assignments spread out over the semester, which are pre-specified. In addition, you can define your own semester programming project for a GPU programming topic that you like. There will be weekly reading assignments, and we will do several quizzes (written; closed-book).
Course Policies	Attendance is mandatory. Every day past the deadline of an assignment reduces the points by 10%.
Additional Information	The course webpage is located here: faculty.kaust.edu.sa/sites/markushadwiger/pages/cs380.aspx This includes the slides from previous years, which should give a good in-depth overview of the contents of the course.

Tentative Course Schedule

(Time, topic/emphasis & resources)

Week	Lectures	Topic
1	Mon 08/26/2019 Wed 08/28/2019	Introduction
2	Mon 09/02/2019 Wed 09/04/2019	GPU Architecture 1 & 2
3	Mon 09/09/2019 Wed 09/11/2019	GPU Architecture 3 & 4
4	Mon 09/16/2019 Wed 09/18/2019	GPU Architecture 5 & 6
5	Mon 09/23/2019 Wed 09/25/2019	Saudi National Day GPU Architecture 7
6	Mon 09/30/2019 Wed 10/02/2019	Shading and Compute APIs 1 & 2
7	Mon 10/07/2019 Wed 10/09/2019	GPU Texturing 1 & 2
8	Mon 10/14/2019 Wed 10/16/2019	GPU Texturing 3 & 4 Virtual Texturing
9	Mon 10/21/2019 Wed 10/23/2019	Stream Computing and GPGPU
10	Mon 10/28/2019 Wed 10/30/2019	Mid-semester break CUDA Memory Access 1
11	Mon 11/04/2019 Wed 11/06/2019	CUDA Memory Access 2 & 3
12	Mon 11/11/2019 Wed 11/13/2019	CUDA Memory Access 4 & 5
13	Mon 11/18/2019 Wed 11/20/2019	Parallel Reduction Parallel Prefix Sum / Scan
14	Mon 11/25/2019 Wed 11/27/2019	Shuffles, Atomics, Instruction Level Parallelism Cooperative Thread Groups
15	Mon 12/02/2019 Wed 12/04/2019	CUDA Unified Memory Sparse Matrix-Vector Ops (SpMV)
16	Mon 12/09/2019	Semester project presentations

Note

The instructor reserves the right to make changes to this syllabus as necessary.