Course title	Code	semester	T+U	credit	ECTS
CUDA Programming		4	3+0	3	4

Prerequisite Courses	None
Language of the Course	English
Course Level	Undergraduate
Type of Course	Optional
Course Coordinator	
Instructors	
Course Assistants	
The aim of lesson	Student will learn about GPU as part of PC architecture. They will then learn how to develop GPU software using CUDA C and OpenCL. Various optimization issues will be discussed. Optimization concepts and their effects will be illustrated with case studies.
Course Content	The course is designed to give hands-on knowledge and development experience in general-purpose GPU programming. Students will learn about GPU as part of PC architecture. They will then learn how to develop GPU software using CUDA C and OpenCL. Various optimization problems, especially efficient use of memory and floating point calculations will be discussed. Optimization concepts and their effects will be illustrated with case studies. Students will be expected to propose a computationally costly problem to implement on the GPU, and then develop and optimize it on the GPU and compare the performance results with the CPU implementation.
Course Learning Outcomes	After completing this course; The student will have in-depth knowledge of parallel programming and parallelism models, the student will be able to compare different GPU programming frameworks. The student will be able to design and develop algorithms on the GPU, and will be able to know and perform optimization techniques.

Weeks	Topics
one	GPU architecture
2	CUDA programming model, thread execution
3	CUDA memory hierarchy
4	Synchronization and reduction
5	Dynamic parallelism and onboard memory
6	Design and optimization of GPU algorithms.
7	Streaming data processing, computation-communication overlap.
8	Multiple GPU systems.
9	Nvidia Thrust library
10	OpenCL basics
11th	OpenCL memory management
12	Code optimization with OpenCL
13	Libraries and tools for GPU programming
14	Application Examples and Events
15	Dynamic Task Sequences and Synchronization

General Competencies

Acquiring algorithm development and application skills for parallel and distributed systems

resources

NVidia, CUDA Programmng Guide, available from http://www.nvidia.com/object/cuda_develop.html for CUDA 2.0 and Windows, Linux or MAC OS.

Grama, A. Gupta, G. Karypis, and V. Kumar, Introduction to Parallel Computing, 2nd Ed. (Addison-Wesley, 2003)

Evaluation System
The dates, days and hours of the Midterm Exam, Quiz, Final Exam and Evaluations will be announced later, according to the decision of the Faculty Administrative Board.

	WITH PROGRAM LEARNING OUTCOMES COURSE LEARNING OUTCOMES RELATIONSHIP TABLE										
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PC10**	**PC11**
INCREASE 1	3	2	one	3	2	2	one	2	2	one	one
INCREASE 2	3	3	2	3	2	one	2	2	3	one	2
INCREASE 3	3	3	2	3	2	one	one	one	2	2	one
LO: Learning Outcomes OP: Program Outcomes											
Contribution Level	**1 Very Low**		**2 Low**		**3 Medium**		**4 High**		**5 Very High**		

Relation of Program Outcomes and Related Course

Lesson	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
Parallel Programming with GPU	3	3	2	3	2	one	1	2	2	1	1