FITCHBURG STATE UNIVERSITY Academic Year 2017–2018 GCE #17-18-14

GRADUATE COUNCIL

NEW COURSE PROPOSAL

Please submit the typed original with all required signatures to the Graduate Council

Course syllabus must be attached and completed according to Fitchburg State guidelines

Form functionality most compatible using Adobe Acrobat 9 and newer.

Course Title: Parallel Programming with CUDA

Banner limit of 30 characters, including punctuation, spaces, and special characters.

Banner Abbreviation:	P	a	r	a	l	l	e	l		P	r	o	g	r	a	m	m	i	n	g		w	i	t	h		C	U	D	A

Faculty member(s) who are creating the course: Brady Chen

Contact Person: Brady Chen

Graduate Program Proposing Change:

Course Description:

This topics course covers programming techniques on the parallel computing architecture and programming model known as Compute Unified Device Architecture (CUDA). The topics include the introduction of CUDA, the basic CUDA commands and syntax, as well as several optimizations for CUDA code and utilization of CUDA libraries.

Students in the course will learn how to develop scalable parallel programs targeting the unique requirements for obtaining high performance on GPUs.

Rationale for Offering the Course:

Parallel programming is rapidly growing technique that uses multi- and/or many-core computer systems to solve a problem at a greater computational speed than using a single computer. This creates a high demand for people with parallel programming skills.

This course will serve two purposes for students in computer science graduate program:

1. **Provides the students with knowledge of parallel computing architecture such as CUDA and parallel programming skills.**
2. **Offer the graduate students an elective course in the area of parallel computing.**

The materials in this proposal have been taught as independent study in the past semesters.

FITCHBURG STATE UNIVERSITY Academic Year 2017–2018 GCE #17-18-_____

GRADUATE COUNCIL
NEW COURSE PROPOSAL

Is there a similar undergraduate course? ● Yes ○ No If so, how does this graduate course differ?

The Both the graduate and undergraduate courses cover similar topics and use same materials and computer lab. The main difference lies in the course projects in both breadth and depth as many of the graduate students have broad knowledge in various disciplines.

Discipline Prefix: CSC

If more than one provide rationale:

Credits: 3

Brief rationale for level choice: The course serves as a graduate elective.

Level: ○ 7000 ● 8000 ○ 9000

Additional Requirements: Laboratory Hours: _____ Pre-Practicum Hours: _____ Practicum Hours: _____

Fieldwork Hours: _____ Other (specify): _____

Prerequisite course(s) if any: Pass the graduate computer science Accuplacer test

This course will be: □ a Required Course ☑ an Elective Special/Note: _____

Course is a replacement for (Course Number/Name): _____

Has the course been offered previously as a "Topics" course? ● Yes ○ No

If yes, How often? Every year

What is the Expected Average Enrollment?: 15

Which semester will this course first be offered?: Spring 2019 How often thereafter to be offered?: Every year

Does this course affect offerings in any other department or program? ○ Yes ● No If yes, please explain.

Is this an Extended Campus course? ○ Yes ● No

FITCHBURG STATE UNIVERSITY Academic Year 2017–2018 GCE #17-18-14

GRADUATE COUNCIL
NEW COURSE PROPOSAL

Course syllabus must be attached and completed according to Fitchburg State guidelines.

☒ Course Syllabus is Attached

Print Form

☑ Reviewed By Dean: Date: 11/27/17

Required Signatures—Graduate and Continuing Education New Course Approval

Course Developer: Brady Date: 11/18/2017

Department Chair: Brady Date: 11/18/2017

Graduate Council Chair: ~~Brady~~ Renée M Scapparone Date: ~~11/18/2017~~ 02/6/2018
(Indicates Graduate Council approval)

President: S. Lapidus Date: 2/21/18

Parallel Programming on GPU 11/28/2017

Instructor: Brady Chen

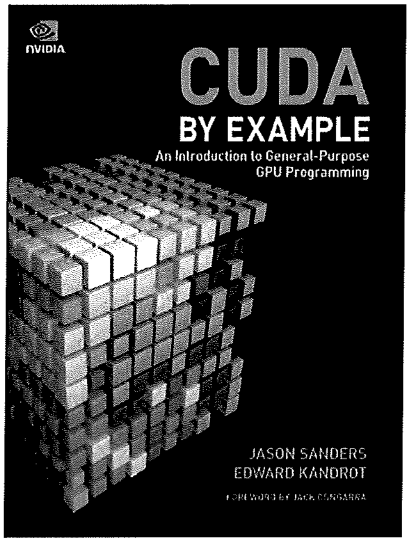
Office and Phone: E310 Edgerly Hall, 978-665-3434

E-mail: xchen@fitchburgstate.edu

Text: CUDA by Example: An Introduction to General-Purpose GPU Programming, 1/E by Sanders & Kandrot. ISBN-10: 0131387685 ISBN-13: 9780131387683

Other resources:

CUDA Toolkit Documentation v6.5
http://docs.nvidia.com/cuda/index.html#axzz3DcnSNeDt
CUDA Getting Started Guide
http://docs.nvidia.com/cuda/cuda-getting-started-guide-for-microsoft-windows/index.html#axzz3DcnSNeDt


NVIDIA
CUDA
BY EXAMPLE
An Introduction to General-Purpose GPU Programming
JASON SANDERS
EDWARD KANDROT
FOREWORD BY JACK DONGARRA

Course Requirements and Grading:

Projects	**40%**
Homework assignments	**25%**
Test	**15%**
Final exam	**20%**

Course Description: This topics course covers programming techniques on the parallel computing architecture and programming model, CUDA. The topics include the introduction of CUDA, the basic CUDA commands and syntax, as well as several optimizations for CUDA code and utilization of CUDA libraries.

Students in the course will learn how to develop scalable parallel programs targeting the unique requirements for obtaining high performance on GPUs.

Prerequisite: CSC 2560.

Topics Covered (Tentative):

- Introduction to CUDA architecture
- GPU hardware and parallel communication patterns
- Parallel Programming in CUDA C
- Thread Cooperation
- Constant Memory and Events
- Texture Memory
- Graphics Interoperability
- Atomics
- Streams
- CUDA C on Multiple GPUs

Final Grade Assignments:

I will calculate the final grades as follows:

Range	< 60	60-61	62-64	65-66	67-69	70-71	72-74
Grade	0.0	0.0	0.0	0.0	0.0	2.0	2.3
Range	75-76	77-79	80-82	83-86	87-89	90-93	> 93
Grade	2.5	2.7	3.0	3.3	3.5	3.7	4.0

The blackboard:

The blackboard is the primary tool used for posting class announcements, grades, assignments, projects, tests, and all class related information. The grade center on blackboard is used for posting your grades only. Please do not use the total points on the blackboard grade center to calculate your final grade.

Programming Policy:

Program must run in order to receive full credit. Partial credit is given if you were not able to incorporate all the required features. Submit the best program that you could compile and run successfully.

Good programming style will account for a substantial portion of the grade assigned to the programming part of the assignments.

Submissions:

Please check the due date for each assignment. In order to be considered 'on-time', assignments must be submitted either in class on the due day or via blackboard before the midnight on the due day. Late submissions may be accepted within a week after the due day. The late penalty is 20%. Assignments that are more than one week late may not be accepted.

If programming is required you must submit both the document and source code electronically.

Tests and exam:

There are no make-ups on tests and final exam unless you are hospitalized or on Jury duty on that day. You get a zero on any missed ones.

Schedule of some tests may change with short notice. You are responsible for all the material covered in the class and all the announcements made in the class.

Collaboration and other Issues:

Discussion of material covered in class is strongly encouraged. It is very acceptable to help or receive help from other students in this class. That's the purpose of the blackboard discussion board.

However, the work you submit must be your own work and must be a solid part of your learning process. You cannot copy others and send it in as yours.

Schedule of tests, exam and assignments may change with short notice. You are responsible for all the material covered in the class and all the announcements made in the class.

Final Exam:

TBD