ECE Special Topic Course Request

1a. Course Title: GPU Architectures

1b. Short Title (< 24 chars. incl. spaces; for transcripts): GPU Architectures

2. Term and Year:
Spring 2018

3. Course Description (25 words or less):
This course provides an in-depth coverage of advanced microarchitecture concepts and performance optimizations for modern general purpose graphics processing unit (GPU) accelerators

4. Level:
☐ 18xx ☐ 28xx
☐ 38xx ☐ 48xx
☒ 88xx (Graduate)

5. Instructor(s) for this offering: Sudhakar Yalamanchil

6. Campus(es) for this offering:
☒ Atlanta
☐ GT-Lorraine
☐ GT-Savannah
☐ GT-SJTU
☐ Video
☐ Other:

7. Credit Hours: ☒ 3-0-3 ☐ 2-3-3 ☐ 3-3-4 ☐ Other:

Laboratory (if applicable): ☐ unscheduled ☐ scheduled

For courses with substantial laboratory or project work, a 2-3-3 or 3-3-4 credit distribution may be appropriate. Institute policy requires all lecture courses (3-0-3, most 2-3-3 or 3-3-4) to have a final exam, given during the scheduled final exam period. For courses with major projects, an alternate final exam format (e.g., oral project presentations) may be approved.

8. Course taught previously? ☐ No ☒ Yes, list terms and enrollments: Fall2015 (29)

9. (GRADUATE only) **MS/PhD course domain(s):**

☐ Bio ☒ CSS ☐ DSP ☐ EDA ☐ Emag ☐ Energy ☐ Micro ☐ Optics ☐ Sys&Ctrl ☐ Telecom ☐ VSDD

Approval must be obtained from ALL TIGs responsible for the specified course domain(s) before submission to ECEGC.

10. Prerequisites: ☐ check here if there are NO prerequisites for this course

ECE 6100 or CS 6290 or equivalent and graduate standing

Use an asterisk to indicate that concurrency is allowed and "[C]" to indicate that a minimum grade of C is required. For example, the prerequisites for ECE 4007 would be listed as follows: **ECE 3042[C] and ECE 4001***

ECE 4xxx courses are considered advanced topics in EE/CmpE and, therefore, generally must have another ECE course as prerequisite. Prerequisites for undergraduate courses are enforced and may not be waived by the instructor, although ECE graduate students will be granted prerequisite overrides. Prerequisites are not enforced for graduate courses.

11. Textbooks: ☐ check here if NO textbooks are to be ordered for this course

For each text to be ordered, specify either "required" or "optional" and provide the other requested information.
To specify more than three texts, include the information for the additional texts on the syllabus and check here: ☐

	Reqd/Opt	ISBN Number	Author, Title, Publisher, Year
a)	Optional	ISBN-13: 978-01238	D. Kirk, and W. Hwu, "Programming Massively Parallel Processors: A Hands-on Approach
b)	Required		
c)	Required		

12. Special Considerations (*e.g., major/level restrictions, cross-listing with another department, non-traditional scheduling)***:**

This form must be submitted each time a course is proposed for offering as a special topic in ECE. To also request **permanent catalog listing**, the ***ECE New Course Request*** form must be completed and submitted along with this form.

In addition to this completed form, the following information is required (unless also included in a permanent listing request):

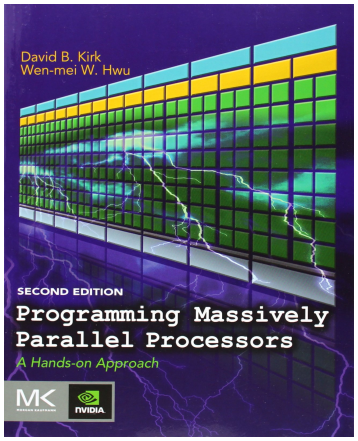
- Course syllabus, including a **1-page** topical outline and a description of how students will be evaluated or graded.
- Summary of grades assigned and CIOS student evaluations for all previous offerings of the course.

ECE Special Topic Course Request

After TIG approval, (1) all materials must be submitted to the chair of the ECE Undergraduate or Graduate Committee, as appropriate, and (2) this completed form and the course syllabus must be submitted **electronically** to Mary Ann Weitnauer.

EE 8823: GPU Architectures

3-0-3 (2S,1D) Prerequisite: EE 6100, CS 6290 or equivalent



The last decade has seen the emergence of general-purpose graphics processing units (GPUs) as vehicles for accelerating general purpose scientific, enterprise, and embedded applications. This emergence has coincided with the explosive growth of data parallel applications and the ascendance of energy efficiency as a driver of performance scalability. The research community has evolved a body of compiler and microarchitecture knowledge to address important bottlenecks to harnessing the enormous throughput and memory bandwidth of modern GPUs. This course first provides an in-depth coverage of important microarchitecture concepts and performance optimizations that have now become accepted in this research and product community. This is followed by coverage of more recent research advances in the performance and power optimization of GPUs.

GPUs are now seeing increasing computation from other models such as Systolic and Dataflow. The course concludes with an exposition of the key elements of these models in contrast t

Class Materials:

- D. Kirk, and W. Hwu, "Programming Massively Parallel Processors: A Hands-on Approach," Morgan Kaufman (pubs), Second Edition, Print Book ISBN: 9780124159921 eBook ISBN: 9780123914187
- Conference and Journal Publications
- Class Notes

Topical Outline

- Introduction
  - Bulk Synchronous Parallel (BSP) models
  - CUDA vs. OpenCL
  - BSP Algorithms for common primitives
- Microarchitecture
  - Basic microarchitecture concepts and the SIMT execution model
    - Kernel launch, scheduling, and control flow management
  - Memory hierarchy operation
    - Memory coalescing and shared memory management
    - Cache management
  - Discrete vs. integrated GPUs
- Control Divergence
  - Introduction to control divergence and solutions
  - Optimizations for control divergence management

        - Dynamic Warp Formation and Thread Frontiers
        - Thread Block Compaction and Dynamic Warp Subdivision
        - Emerging techniques
- GPU Memory Hierarchy: Key concepts underscoring the operation of memory hierarchies in discrete and integrated GPUs
    - Uniform virtual memory (UVM)
    - CPU- GPU coherency issues
    - Introduction to memory divergence and latency hiding techniques
    - Dynamic vs. static techniques for mitigating memory divergence
- Scheduling: Scheduling optimizations
    - Warp scheduling algorithms
    - Thread block scheduling
    - Optimizations for throughput vs. energy
- Advanced Microarchitecture Concepts: Optimizations of the GPU microarchitecture and memory system
    - Organization of the register files and RF-Core interconnect
    - Cache and memory system optimizations
    - Optimizations for power and energy efficiency
- Competing (Accelerator) Models and Architectures
    - Systolic Model of Computing
        - Basic model elements and algorithmic primitives
    - Data Flow Execution
        - Static and dynamic data flow

Course Grading:

Mid-term: 15%
Assignments: Mini-projects (40%)
Final Project: 35%
Final: 10%

The bulk of the course content will be based on material from workshop, conference, and journal publications. The midterm will test understanding of fundamental concepts. The project will address a coherent research theme in modern GPU architectures. The final exam will be based on a project report in conference paper format.

Georgia Institute of Technology

Fall 2015, ECE 8823 Special Topics Section A
Instructor: Yalamanchili, Sudhakar (Primary)

There were: 30 possible respondents.

	Question Text	N	RR	Interpol. Median	0-3	3-6	6-9	9-12	12-15	15-18	18 +	N/A
1	Student: Hours per week	18	60%		0	1	7	4	3	1	2	0
					0-30	30-50	50-70	70-80	80-90	90-100	N/A	
2	Student: Percent attendance	17	57%		0	0	1	1	5	10	0	
3	Student: Percent homework completion	18	60%		0	0	0	0	2	16	0	
					5 Extremely Well	4	3	2	1 Completely Unprep	N/A		
5	Course: How prepared to take subject	18	60%	3.8	2	10	4	1	1	0		
					5 Exceptional Amt	4	3	2	1 Almost Nothing	N/A		
6	Course: Amount learned	18	60%	3.8	4	7	6	1	0	0		
					5 Exceptional	4	3	2	1 Very Poor	N/A		
7	Course: Assignments facilitated learning	18	60%	3.9	3	10	4	1	0	0		
8	Course: Assignments measured knowledge	18	60%	3.7	2	9	4	3	0	0		
					5 Strongly Agree	4	3	2	1 Strongly Disagree	N/A		
9	Course: Overall effectiveness	18	60%	4.2	6	9	2	0	1	0		
					5 Exceptional	4	3	2	1 Very Poor	N/A		
14	Instructor: Clarity (Yalamanchili)	18	60%	4.1	7	5	5	0	1	0		
					5 Strongly Agree	4	3	2	1 Strongly Disagree	N/A		
15	Instructor: Communicated how to succeed (Yalamanchili)	18	60%	4.5	9	5	3	1	0	0		
					5 Exceptional	4	3	2	1 Very Poor	N/A		
16	Instructor: Respect for students (Yalamanchili)	18	60%	4.4	8	7	3	0	0	0		
					5 Extremely Enthus	4	3	2	1 Detached	N/A		
17	Instructor: Enthusiasm (Yalamanchili)	18	60%	4.5	9	4	5	0	0	0		
					5 Made Me Eager	4	3	2	1 Ruined Interest	N/A		
18	Instructor: Stimulates interest (Yalamanchili)	18	60%	3.9	6	5	6	1	0	0		
					5 Highly Accessible	4	3	2	1 Hard To Find	N/A		
19	Instructor: Availability (Yalamanchili)	18	60%	4.1	5	9	1	1	1	1		
					5 Extremely Helpful	4	3	2	1 Not Helpful	N/A		
20	Instructor: Feedback helpfulness (Yalamanchili)	18	60%	3.6	5	4	5	3	0	1		
					5 Strongly Agree	4	3	2	1 Strongly Disagree	N/A		
21	Instructor: Overall effectiveness (Yalamanchili)	18	60%	4.2	7	7	3	0	1	0		

Instructor	Text Responses
	Question: Course best aspect
	The assignments were very 'real world.' We built real applications that were directly relevant to our learning.
	Learning about a lot of recent research
	Section after mid term
	The projects were good in intention but poorly structured and TA did not help much either.
	None
	The novelty and the research prospects and the open endedness of the course is amazing. It offers an enormous research potential.
	Question: Course improvements
	A wee bit more preparation. But that is understandable, being the first time the class was offered.
	Assignments could have more cuda
	The lectures should be distributed more among the different topics rather than just focusing on one (control divergence).

	Section before mid term can be slightly altered. Later section and papers were interesting.
	Please give a better description of the project, I wrote a 2000 line code for a project description of one page. Also, the testing benchmark testing time has to be communicated. PACE is a poor cluster to test and run simple C++ code, I would like it to be more flexible. Please give your own makefile and interface classes and driver code. Eases up my work to make TA understand what I did and why my code works.
	Better professor, better TA, better assignments
	Kindly improve the assignments. The reference profiler (HARP) is not very efficient in helping us achieve what is expected out of us. Kindly increase the walltime on the PACE machines that students get, and also the space alloted to each student. Even a single trace takes like 15-20 GB of space. And thus we are not able to generate reference results for help. KINDLY INCREASE THE RESOURCES ALLOTED TO STUDENTS ON PACE (disk memory as well as walltime) - atleast 75-100GB of space allotment is needed, for the humongous trace sizes per binary.
	Question: Instructor greatest strength
Yalamanchili	Profound understanding and interest in what he was teaching.
Yalamanchili	Professor structured this course nicely, but the lack of classroom preparedness took the interest off from the course.
	Question: Instructor improvements
Yalamanchili	A little more curriculum. It's hard to get a lot out of conference papers.
Yalamanchili	Please be well prepared before the class, I takes out all the interest when the professor is not that well prepared.
Yalamanchili	More resources needed on PACE. TAs with more clarity on concepts and lab. More involvement from TA on resolving queries.