

Parallel Programming University Course Materials Development Proposal

Undertaken by

UNIVERSITY and NVIDIA

1 *UNIVERSITY* Services Overview

1.1 Description

Create an original set of adaptable and modifiable course materials for one or more NVIDIA GPU-based university courses on Parallel Programming, with enough materials, detail, and information for use in both an introductory course and an advanced course. Some of the course materials will be generic parallel programming material that can be easily adapted to most multicore processor resources. Most of the materials will scale to different NVIDIA GPU resources, and there will be instructions on how to use and access the *Web-based tool* via a new NVIDIA Open edX website for lab deliverables and local CUDA Toolkit command-line tools for larger coding projects deliverables. *Web-based tool*/Open edX will connect on the back-end to NVIDIA's Amazon AWS G2 resources.

Optional deliverables to be confirmed at a later time are a version of *Web-based tool* modified for use with NVIDIA Open edX website, website and an electronic book in ePUB format, identical in form and content to the *textbook*.

1.2 Schedule

Start of <i>UNIVERSITY</i> Services	Effective Date (TBD)
Conference calls with NVIDIA on progress updates, bi-directional feedback, materials review and technical support as needed	Every Wednesday from Effective Date
Respective site visits to meet about planning, updates, bi-directional feedback, materials review and technical support as needed	As Needed
Reasonable test usage of introductory and advanced materials in <i>UNIVERSITY</i> 's Fall 2015 Applied Parallel Programming course	August 24 th , 2015 – December 18 th , 2015
End of <i>UNIVERSITY</i> services, having delivered all materials except e-book conversion	April 6 th , 2016

2 NVIDIA Deliverables and NVIDIA Delivery Schedule

2.1 Description

NVIDIA Deliverable	Description
....	...

2.2 Delivery Dates

NVIDIA Deliverable	Delivery Date
....	...

3 *UNIVERSITY* Deliverables and *UNIVERSITY* Delivery Schedule

3.1 Description

The deliverables should all be adaptable and modifiable materials a university instructor would need to teach an introductory or advanced 4-month academic term course using the *Web-based tool* via a new NVIDIA Open edX website for lab deliverables and local CUDA Toolkit command-line tools for larger coding projects deliverables. *Web-based tool*/Open edX will connect on the back-end to NVIDIA's Amazon AWS G2 resources..

Optional deliverables to be confirmed at a later date based on budgets and development time are also listed below.

A summary of all of the *UNIVERSITY* deliverables is set out in the following table, and a more detailed description of for each *UNIVERSITY* deliverable follows this table.

Deliverable Code	<i>UNIVERSITY</i> Deliverable	Description
WGPUedX	Modified <i>Web-based tool</i> for use with NVIDIA Open edX website	One (1) modified version of <i>Web-based tool</i> web-based GPU programming environment, connected to NVIDIA's AWS resource, as target development platform for all Le and L deliverables for use with NVIDIA Open edX website
S	Syllabus	One (1) electronic document outlining all modules and partition of introductory and advanced course with inline pointers to individual files in directory
LSe	Lecture Slides	Five (6) total sets of slides (.ppt) with detailed lecture notes for in-class lectures for introductory and advanced course for instructor evaluation
LS	Lecture Slides	Electronic slide sets (.ppt) with detailed lecture notes for in-class lectures for every module of introductory and advanced course
Qe	Quizzes and Solutions	Five (6) total electronic documents (Word) containing shorter student multiple choice questions and answers for modules from introductory and advanced course for instructor evaluation
Q	Quizzes and Solutions	Electronic document (Word) containing shorter student multiple choice questions and answers for all modules of introductory and advanced course
Le	Lab Exercises and Solutions	Set of all working demo source code, electronic documents (Word) containing intermediate student lab exercises tied to demo code and answers/solutions, for appropriate evaluation modules from introductory and advanced course for instructor evaluation
L	Lab Exercises and Solutions	Set of all working demo source code, electronic documents (Word) containing intermediate student lab exercises tied to demo code and answers/solutions, for all appropriate modules of introductory and advanced course
Pe	3-4 Week Coding Projects and Solutions	Five (5) total electronic documents (Word) containing 3-4 week student programming project specifications and working project solution source code, not tied to any particular modules
P	3-4 Week Coding Projects and Solutions	Fifteen (15) total electronic documents (Word) containing 3-4 week student programming project specifications and working project solution source code, not tied to any particular modules
Ae	Lecture Audio/Video Recordings	Six (6) electronic audio/video recording files (in high quality, suitable file format) for every slide with electronic (Word) transcripts of LSe for instructor evaluation
A	Lecture Audio/Video Recordings	Electronic audio/video recording files (in high quality, suitable file format) for every slide with electronic (Word) transcripts of LS
UM	Updated Materials	Updated and modified introductory and advanced course materials based on issues, student feedback and instructor experience based on using the teaching materials in <i>UNIVERSITY</i> 's Fall 2015 Applied Parallel Programming course
Optional		

Deliverables:		
EB	E-book	Electronic book in ePUB format, identical in form and content to the <i>textbook</i>

The outline for which the *UNIVERSITY* Deliverables are to be based on is the following:

Module	Topics
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	

12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
	Where to Find Support

Indicates Evaluation Kit modules

Indicates slide decks to be developed

---- Create Introductory and Advanced partitions of Modules? ----

3.2 Delivery Dates

UNIVERSITY Deliverables	Delivery Date
Syllabus and evaluation teaching material (S, LSe, Qe, Le, Pe, Ae for Modules 1,2,3,7,11 and 23)	November 16th, 2015
All updated/modified material based on test run in <i>UNIVERSITY's</i> Fall 2015 Applied Parallel Programming course (UM)	April 6th, 2016
Optional publication of e-book conversion (EB)	One month from 3rd edition of <i>textbook</i> publication date

3.3 Detailed Deliverable Descriptions and Requirements

- WGPUedX
 - o Modified version of *Web-based tool* web-based GPU programming environment, connected to NVIDIA's AWS resource, as target development platform for all Le and L deliverables for use with NVIDIA Open edX website
 - o Allows instructor to provide unique Teaching Kit serial number to *Web-based tool* to create *Web-based tool* "Instructor Account"
 - o "Instructor Account" holders can invite students to register for "Student Account" via email
 - o Pre-loaded with labs/code, information, instructions and associated questions (Le and L, based on delivery dates above), same format as existing *Web-based tool* application
 - o Instructor Account holders can edit labs/code, information, instructions and associated questions, but not change the basic format as existing *Web-based tool* application
 - o Same auto-grading functionality, but with results sent only to Instructor Account
- Syllabus (S)
 - o Introductory and Advanced course partitions covering all modules in module list in Section 3.1
 - o Outlines the structure and content of all modules, deliverables and aligns as much as possible with *textbook*
 - o Inline pointers to individual files in directory
 - o Allow students to comprehend the structure and flow of the introductory and advanced courses
- Lecture Slides (LSe, LS)
 - o Introductory and Advanced course partitions covering all modules in module list in Section 3.1
 - o Substantially conforms to the structure and content of all modules, deliverables, and aligns as much as possible with *textbook*
 - o Covers standard 3-hr/week, 4-month courses
 - o Clear and detailed lecture notes on every appropriate slide a lecturer could use to speak properly to lecture slides
 - o Clear and readable, and reasonably easy to comprehend
- Quizzes and Solutions (Qe, Q)
 - o Introductory and Advanced course partitions covering all modules in module list in Section 3.1
 - o Substantially conforms to the structure and content of all modules, deliverables, and aligns as much as possible with *textbook*
 - o Reasonably challenging and thought provoking, multiple choice questions and should allow for students to display knowledge learned in lecture and textbook materials, as well as develop and test their comprehension and problem-solving skills
 - o Scalable in that they can be somewhat easily changed so that same problems and solutions aren't used every semester to deter cheating
 - o Include clearly-marked solutions for instructors to easily deploy in courses

- Quizzes should be largely different than the *textbook*, about 50% different
- Lab Exercises and Solutions (Le, E)
 - Introductory and Advanced course partitions covering all appropriate modules in module list in Section 3.1
 - Substantially conforms to the structure and content of all modules, deliverables, and aligns as much as possible with *textbook*
 - Reasonably challenging and thought provoking, and should allow for students to display knowledge learned in textbook and lecture materials, as well as develop and test their computer science, engineering, programming and problem-solving skills
 - Scalable in that they can be somewhat easily changed so that same problems and solutions aren't used every semester to deter cheating
 - Include fully-working code, all necessary files and basic instructions on how to run the labs as appropriate on *Web-based tool*/Open edX AWS resource
 - Include clearly-marked solutions for instructors to easily deploy in courses
 - Labs should be largely different than the *textbook*, about 50% different
 - Labs should also contain reference and background information
 - At least one piece of working demo code for each appropriate module
- 3-4 Week Coding Projects and Solutions (Pe, P)
 - Reasonably challenging and thought provoking, and should allow for students to display knowledge learned in textbook and lecture materials, as well as develop and test their computer science, engineering, programming and problem-solving skills
 - Scalable in that they can be somewhat easily changed so that same problems and solutions aren't used every semester to deter cheating
 - Larger weekly programming assignments that would take a student about 3-4 weeks to complete on average
 - Include fully-working projects, source code, all necessary files and basic instructions on how to run the projects using local CUDA Toolkit where appropriate
- Lecture Audio/Video Recordings (Ae, A)
 - Introductory and Advanced course partitions covering all modules in module list in Section 3.1
 - Clear, concise, and understandable
 - High quality electronic formats
 - Broken out into separate recordings, one for each module
 - Including electronic (Word) transcripts for all audio
 - Can be recorded either within real classroom setting or not
- Optional E-book (EB)
 - Identical in form and content to the *textbook*
 - Contingent upon NVIDIA, *UNIVERSITY* and *publisher* agreement and payment of any additional, relevant fees