Literacy for All in Parallel and Distributed Computing (PDC): NSF/IEEE-TCPP Guidelines for an Undergraduate Core Curriculum

Sushil K Prasad
Georgia State University
Former Chair, IEEE Technical Committee on Parallel Processing (TCPP)

South Carolina CyberInfrastructure Symposium
February 11-13, 2013
Clemson University, Clemson, South Carolina

Curriculum Initiative Website:

http://www.cs.gsu.edu/~tcpp/curriculum/index.php

Linked through TCPP site: tcpp.computer.org

Outline

- Why and what are the opportunities for the audience?
- Key Activities and Milestones
  - Planning Workshops, Tele-meetings, EduPar, Early adopter competitions, Interface with 2013 ACM/IEEE CS curriculum Taskforce, CEDR center funding
- How was the curriculum formulated?
  - Blooms classification, learning outcomes, hours, which courses, how to teach
- How is it getting evaluated?
  - Range of courses and early adopter institutions, feedback and resources, EduPar posters and talks
- Lessons learnt
  - How can you restructure your own core courses?
- Conclusions and Roadmap
  - Periodic curriculum update – Preliminary version 2010; first version Dec 2012

Who are we?

- Chtchelkanova, Almadena - NSF
- Dehne, Frank - University of Carleton, Canada
- Gouda, Mohamed - University of Texas, Austin, NSF
- Gupta, Anshul - IBM T.J. Watson Research Center
- JaJa, Joseph - University of Maryland
- Kant, Krishna - NSF, Intel
- La Salle, Anita - NSF
- LeBlanc, Richard, University of Seattle
- Lumsdaine, Andrew - Indiana University
- Padua, David- University of Illinois at Urbana-Champaign
- Parashar, Manish- Rutgers
- Prasad, Sushil- Georgia State University
- Prasanna, Viktor- University of Southern California
- Robert, Yves- INRIA, France
- Rosenberg, Arnold- Northeastern and Colorado State University
- Sahni, Sartaj- University of Florida
- Shirazi, Behrooz- Washington State University
- Sussman, Alan - University of Maryland
- Weems, Chip, University of Massachussets
- Wu, Jie - Temple University

Why now?

- Computing Landscape has changed
  - Mass marketing of multi-cores
  - General purpose GPUs even in laptops (and handhelds)
- A student with even a Bachelors in Computer Science (CS) or Computer Engineering (CE) must acquire skill sets to develop parallel software
  - No longer instruction in parallel and distributed computing primarily for research or high-end specialized computing
  - Industry is filling the curriculum gap with their preferred hardware/software platforms and "training" curriculums as alternatives with an eye toward mass market.

Stakeholders

- CS/CE Students
- Educators – teaching core courses as well as PDC electives
- Universities and Colleges
- Employers
- Developers
- Vendors
- Authors
- Researchers
- NSF and other funding agencies
- IEEE Technical Committees/Societies, ACM SIGs,
- ACM/IEEE Curriculum Task Force

Current State of Practice

- Students and Educators
  - CS/CE students have no well-defined expectation of what skill set in parallel/distributed computing (PDC) they must graduate with.
  - Educators teaching PDC courses struggle to choose topics, language, software/hardware platform, and balance of theory, algorithm, architecture, programming techniques…
  - Textbooks selection has increasingly become problematic each year, as authors cannot keep up; no single book seems sufficient
  - Industry promotes whatever best suits their latest hardware/software platforms.
  - The big picture is getting extremely difficult to capture.

Why did the community and experts get onboard?

- Timing and Community Need
- Everyone is a stakeholder
- Transparency, inclusive
- Community outreach at all stages
- Thoroughness and quality
- Continual Feedback mechanisms
  - From experts and stakeholders
  - Early adopters
  - EduPar workshops
  - Curriculum Sessions/Panel
    - HiPC Dec 2010, India Goa
    - SiGCSE March 2011, Dallas
    - EduPar-11, Alaska
    - EduPar-12, Shanghai
    - SC-12 – invited talk and panel
- Work in progress – opportunity to participate and contribute!

Curriculum Planning Workshops at DC (Feb-10) and at Atlanta (April-10)

- Goals
  - setup mechanism and processes which would provide periodic curricular guidelines
  - employ the mechanism to develop sample curriculums
- Agenda:
  - Review and Scope
  - Formulate Mechanism and Processes
  - Preliminary Curriculum Planning
    - Core Curriculum
    - Introductory and advanced courses
  - Impact Assessment and Evaluation Plan

Main Outcomes

- Priority:
Core curriculum revision at undergraduate level

- Preliminary Core Curriculum Topics

-Sample Intro and Advanced Course Curriculums

Weekly Tele-Meetings on Core Curriculum (May-Dec'10; Aug'11-Feb'12)

Goal: Propose core curriculum for CS/CS graduates

- **Every individual** CS/CE undergraduate must be at the proposed level of knowledge as a result of their *required* coursework

Preliminary version: Dec 2010

Process: For each topic and subtopic

1. Assign **Bloom's classification**

   K= Know the term (basic literacy)
   C = Comprehend so as to paraphrase/ illustrate
   A = Apply it in some way (requires operational command)

2. Write **learning outcomes**
3. Identify core CS/CE courses impacted
4. Assign number of hours
5. Write suggestions for "how to teach"

4 Curriculum Areas

Architecture, Programming, Algorithms, Cross-cutting

Curriculum Example

Algorithms Topics			Bloom#	Course	Learning Outcome
Algorithmic problems					*The important thing here is to emphasize the parallel/distributed aspects of the topic*
	Communication				
		broadcast	C/A	Data Struc/Algo	*represents method of exchanging information - one-to-all broadcast (by recursive doubling)*
		multicast	K/C	Data Struc/Algo	*Illustrate macro-communications on rings, 2D-grids and trees*
		scatter/gather	C/A	*Data Structures/Algorithms*	
		gossip	N	*Not in core*	
	Asynchrony		K	CS2	*asynchrony as exhibited on a distributed platform, existence of race conditions*
	Synchronization		K	CS2, Data Struc/Algo	*aware of methods of controlling race condition,*
	Sorting		C	CS2, Data Struc/Algo	*parallel merge sort,*
	Selection		K	CS2, Data Struc/Algo	*min/max, know that selection can be accomplished by sorting*

Early Adopter Program

- Total 80 institutions worldwide
  - Spring-11: 16 institutions ; Fall'11: 18;
  - Spring-12: 21; Fall-12: 25 institutions
  - Most from US (4 year to research institutions);
    - some from South America, A few from Europe, fewer from Asia
    - Wittenberg University and Clemson University – Spring 2011
- **Fall-13 round of competition:** Deadline June 30, 2013
  - NSF funded Cash Award/Stipend up to $2500/proposal
  - *Which course(s) , topics, evaluation plan?*
- **Instructors for core CS/CS courses** such as CS1/2, Systems, Data Structures and Algorithms – department-wide multi-course multi-semester adoption preferred
  - Elective courses; graduate courses

Courses updated by Early Adopters - Fall 2012

- Swarthmore College
  - CS31 Introduction to **computer systems**, CS40 Computer graphics, CS41 algorithms, CS45 Operating Systems, and CS87 Parallel computing
- Oklahoma City University
  - **CS1, CS2, CS3, Software Engineering**
- Singapore University of Technology and Design
  - Introduction to **Algorithms - departmental core**; Computer **System Engineering - departmental core;** and Graph Theory and Algorithm - elective.
- Purdue University
  - **ECE 264 - Advanced C Programming**
- Huazhong University of Science and Technology
  - Parallel Programming Principle and Practice; Parallel **Data Structure and Algorithm**
- Wilberforce University
  - IDS L, **university wide core**, and a co-op course.
- Carnegie Mellon University & Pittsburgh Supercomputing Center
  - Introduction to **Computational Physics** and Advance Computational Physics
- Louisiana State University
  - **Digital Logic** I, II

EduPar Workshop Series

- EduPar-11 at Alaska, IPDPS-2011
  - Receive feedback from the Adopters
  - Stimulate discussion of curricular and other educational issues.
- EduPar-12 at Shanghai, IPDPS-2012
  - A regular satellite workshop
  - Selected 20 of 50 early adopters to attend
- *EduPar-13* will be at Boston in May 2013

Current Activities

- Curriculum Revision and Formal Curriculum Release
  - Revision through Fall 2011 and Spring/Summer 2012
  - Formal release in Dec 2012
- Educational Resource Website
  - Call for contribution
- Book project:
  - Part 1: Guide for Instructors
  - Part 2: Resource for students
- Interface to the Broader Community
  - ACM/IEEE taskforce for CS Curriculum revision CS-2013.

Lessons learnt

How can you restructure your own core courses?

Data Structures and Algorithms (DS/A) course

- Sampled over 3 courses

Algorithms - Topics	DS/A Totals for Topics			
	K	C	A	Average
Asymptoptic	0	0	6	2
Time	0	0	3	1
Space	0	1	1	0.7
Speedup	0	0	0	0
Cost Reduction:	0	0	1	0.3
Space Compression,etc	0	0	1	0.3
Time vs Space	0	2	2	1.3
Power vs Time, etc.	0	3	0	1
Scalability in Algorithms and Architectures	0	2	1	1
Notions from Complexity-Theory:	2	0	0	0.7
P-Completelness	2	0	0	0.7
#P-Completeness	1	0	0	0.3
Cellular Automata	2	0	0	0.7
Dependencies	1	0	0	0.3

Algorithms - Topics	DS/A Totals for Topics			
	K	C	A	Average
Divide and Conquer (Parallel Aspects)	0	2	1	1
Recursion (Parallel Aspects)	0	1	3	1.3
Scan (Parallel Prefix)	0	0	1	0.3
Reduction	0	0	1	0.3
Map-Reduce	0	0	1	0.3
Series-Parallel Composition	0	0	0.5	0.2
Asynchrony	0.5	0	0	0.2
Sorting	0	2	4	2
Selection	0	2	3	1.7
Graph Algorithms	0	2	2	1.3
Search	0	3	2	1.7
Path Selection	0	4	0	1.3
Specialized Computations	0	0	3	1
Matrix Computations	0	0	1	0.3
Matrix Product	0	0	0.5	0.2
Hour totals per Course	2	6	9	6

Data Structures and Algorithms (DS/A) course

Crosscutting - Topics	DS/A Totals for Topics			
	K	C	A	Average
Why and What is Parallel/Dist Computiting	0	3	0	1
Concurrency	0	1.5	0	0.5
Power	1	0	0	0.3
Web Search	0	2	0	0.7
Social Network/ Context	0	1	0	0.3
Hour totals per Course	.25	2	0	0.7

Programming - Topics	DS/A Totals for Topics			
	K	C	A	Average
Shared Memory	0	0.5	0	0.2
Distributed Memory	0.5	0.5	0	0.3
SPMD	0	0	1	0.3
Data Parallel	1	0	0	0.3
CUDA/OpenCL	0	0	1	0.3
Computation	0	0	2	0.7
Computation Decomposition Strategies	0	2	2	1.3
Decomposition into atomic tasks	0	1	0	0.3
Static	0	1	0	0.3
Dynamic	0	1	0	0.3
Data	0	0	4	1.3
Performance Monitoring	0	1	0	0.3
Speedup	0	1	0	0.3
Efficiency	0	1	0	0.3
Hour totals per Course	0.4	2	2.5	1.6

Syllabus for a sample Data Structure and Algorithms

(Second semester with only programming as their prior background, IIIT Hyderbad, India)

1. Introduction to data structures (Week 1)
2. Introduction to asymptotic analysis – mainly O(.) notation (Week 2)
3. Array as a data structure, sorting, parallel sorting, parallel prefix (Week 3)
4. Stacks and queues (Week 4)
5. Linked lists, ideas from list representation and ranking and its difficulty in the parallel setting. (Week 5)
6. Trees, applications to evaluation, searching, balanced search trees, scope for parallel operations (Week 6-7)
7. Graph traversal techniques, shortest paths, spanning trees, solutions in the parallel setting (Week 8-10)
8. Advanced data structures such as Union-Find, B-trees, Suffix tree, trie (Week 11-13)

Center for Parallel and Distributed Computing Curriculum Development and Educational Resources (CDER)

- Develop PDC core curricula flexible enough for a broad range of programs and institutions; collaborate with all stakeholders
- Develop, collect, and synthesize pedagogical and instructional materials for teaching PDC curriculum topics*
- Facilitate access to state-of-the-art hardware and software resources for PDC instruction and training by instructors and students*
- Organize Early Adopter Competitions and EduPar workshops, and related events*

* Call for participation and contribution

Conclusion

- Time is right for PDC curriculum standards
- Core Curriculum Revision is a community effort
  - **Curriculum Initiative Website:**
  - http://www.cs.gsu.edu/~tcpp/curriculum/index.php
  - Linked through TCPP site: tcpp.computer.org

- *Email sprasad@gsu.edu*
- *Need to inculcate "parallel thinking" to all*

Acknowledgements

- NSF: Primary Sponsor
- Intel: Early Adopters
- IBM: EduPar Workshop
- NVIDIA: Early Adopters

Literacy for All in Parallel and Distributed Computing (PDC): NSF/IEEE-TCPP Guidelines for an Undergraduate Core Curriculum

Sushil K Prasad
Georgia State University
Former Chair, IEEE Technical Committee on Parallel Processing (TCPP)

South Carolina CyberInfrastructure Symposium
February 11-13, 2013
Clemson University, Clemson, South Carolina

Curriculum Initiative Website:

http://www.cs.gsu.edu/~tcpp/curriculum/index.php

Linked through TCPP site: tcpp.computer.org