A Planning Workshop on Curriculum Standards for Parallel and Distributed Computing

1. Summary

There is an urgent need for curricular guidance on parallel and distributed computing. This need emanates both from rapid technological changes and from mass marketing of multicores and general-purpose graphics processing units. Educators struggle to decide the right balance between theory and practice, and to choose from the diverse set of models of computation, languages, software and hardware platforms, and tools. There are other stakeholders including students, employers, researchers, authors, developers, users, and industry who all can benefit from periodic standards in curriculum at various levels and in different courses which are impacted from parallel and distributed computing.

This workshop is to explore the state of curriculum in parallel and distributed education, assess the needs, and provide action plan and recommend mechanisms for how best to address the curricular needs in short and long term. The planning workshop and its related set of activities will draw experts from various stakeholders, and engage the parallel and distributed computing community as providers of information on the current state of practice as well as consumers and evaluator of its results. The main benefit would be to students in preparation for college and graduate schools, professional and research career, and re-training. These curricular guidance and its trajectory, along with their periodic feedback and other evaluation data on their adoption and use will also help steer companies hiring students and interns, hardware and software vendors, and, of course, the authors and researchers.

2. Statement of Objectives

The goal of this initiative is to assess the state of practice, needs, and setup mechanism and processes which would provide periodic curricular guidelines for educators (and their students) on
- their introductory and other courses on parallel and distributed computing at undergraduate and graduate levels,
- relevant modules for other computer and information science and engineering basic courses such as introductory set of programming and discrete math classes, and advanced courses such as architectures, algorithms, programming languages, operating systems, networking, graphics and visualization, biomedical computing, scientific computing, etc., and
- possibly relevant modules for K-12 AP and IB classes, which will have "computational thinking" component, and for retraining and/or certification of current computing professionals.
3. Statement of Need and List of Topics

3.1 Current State, Need and Vision

Parallel and Distributed Computing now permeates most computing activities. The penetration of this technology in the daily lives has resulted in common users relying on its effectiveness and reliability. The mass marketing of multicores and general-purpose graphics processing units (GP-GPUs) in home and office PCs and laptops (and, to a certain extent, the proliferation of handhelds and smart phones increasingly capable of inter-device computation) has a potential for empowering even common users to become a technology contributor. Certainly, it is no longer sufficient for even basic programmers to acquire only the conventional programming skills. All this phenomena point to the need for imparting a broad-based skill set in parallel and distributed computing at various levels, impacting undergraduate and graduate Computer Science/Engineering programs and related computational disciplines, and possibly the K-12 Advanced Placement (AP) and International Baccalaureate (IB) CS courses.

However, the rapid change in computing hardware platforms and devices, languages, and supporting programming environments accompanied by their diversity, and the research advances, more than ever challenges the educators what to teach in any given semester. Students and their employer face similar challenges on what constitutes basic (and advanced level) expertise. Currently, professors essentially teach what they are most comfortable with and use whatever computing platform is accessible. In an attempt to keep up with the latest one perhaps ends up ignoring more fundamental aspects. The authors of textbooks simply are not able to keep up, and no single book seems sufficient. Industry promotes whatever best suits their latest hardware/software platforms. The big picture is getting extremely difficult to capture. In short, the entire parallel processing community, facing this rapid churn in technology, is in urgent need of regular, periodic curricular guidance.

Our vision is all stakeholder experts working together and annually/biannually providing guidance on restructuring standard curriculum across various courses and modules related to parallel and distributed computing. The community will employ these for teaching, for producing course material, textbooks and tutorials, setting standards for certifications and exams, and for creating programming environments and tools.

The proposed planning workshop is to bring in experts from all sides to further explore the needs and current problems, assess the state of parallel processing education across curriculum, and lay out a long term plan on setting up mechanisms to meet the curricular challenges. Two outcomes expected are an annual curricular workshop at IPDPS, TCPP’s flagship conference (2010 being held in Atlanta in April) and formation of TCPP’s Standards Committee on Curriculum.

3.2 List of Topics

i. Review of State of Parallel/Distributed Computing Education
   1. Need and Scope Assessment
   2. Stakeholders and how to engage them all
3. Current curriculums
4. Current Supporting infrastructure
   a. Model courses, book series, tutorials, student resources (IDE, debugging tools, software, hardware)
   b. Certifications for Industry Personnel
   c. Student Programming Competitions and other Forums

ii. Courses/Levels to be Considered
   1. Undergraduate courses
      a. Introductory course(s) in parallel and distributed computing
      b. Related courses – OS, Architecture, Computational Math/Sciences, Networks, Software Engineering, Simulation, VLSI, introductory programming courses, etc.
      c. Other courses – Database, Graphics, AI, etc.
      d. Concentrations, minors
   2. Graduate courses
   3. Other levels/disciplines (AP Computer Science, IB CS; Computational Thinking across curriculum at k-12, other disciplines -arts, music, business, law, medicine)

iii. Annual workshop at IPDPS
    - Scope, agenda, format, organization

iv. TCPP Standards Committee on Curriculum
   - Membership drawn from all stakeholders
   - Role, Activities

v. Impact Assessment and Evaluation Plans