Adopting the NSF/TCPP Initiative in Core Systems Education at Portland State University

Karen L. Karavanic
Portland State University
karavanic@cs.pdx.edu

Our efforts to integrate the TCPP Core Curriculum guidelines into two existing PSU systems courses, CS201 Systems Programming and CS333 Introduction to Operating Systems.

Goals

- Integrate the TCPP Core Curriculum guidelines into the undergraduate systems curriculum at Portland State University
- Develop independent course modules for use in CS 201 Computer Systems Programming and CS 333: Introduction to Operating Systems
- Develop new course: Introduction to Performance Measurement, Modeling, and Analysis
- Work with Portland Community College to transfer curriculum revisions
- Review the required systems offerings to fit in new materials

Our Environment

- Portland State University
  - Urban University in Portland, OR
  - ~30,000 students
  - Strong ties with Portland Community College System
  - Located near Oregon’s Silicon Forest, Intel’s Largest Campus
  - On the Academic Quarter System: 3 10-week quarters in the regular academic year
- Computer Science
  - Offers Minor, B.S., M.S., Ph.D., New Beginnings
  - Focus of this project: ABET-accredited B.S. in Computer Science
  - 26 FT faculty
  - 550 undergrad majors
  - ~30% of CS Majors are transfers from Portland Community College
  - Many Post-baccalaureate students: working on their second bachelor’s degree to retrain into computer science
- Curriculum Planning
  - Students come into the required systems courses from a variety of streams: continuing, transfer, post-bac
  - We must develop review modules to transition students into our stream of parallel- and distributed- enhanced systems courses

Project Status

- We are currently developing a new course: Introduction to Performance Measurement, Modeling and Analysis. To be offered Fall and Winter 2014. Modules for use in the required courses will be developed through this course offering.
- Higher level analysis of the systems curriculum is in progress.
- Coordination with Portland Community College not yet started.

Acknowledgments

This work was supported in part by funding from the NSF/TCPP Early Adopters Program, NSF Award #1044973, and NVIDIA corporation.

CS 201: Computer Systems Programming

- A 10-week required course surveying computer organization, data representation, x86 assembly, performance, and the relationship between source code and runtime program execution
- Textbook: Bryant, O’Hallaron, Computer Systems: A Programmer’s Perspective
- Students start lab exercises in lab sessions during class hours, then complete them independently
- Students utilize Lab Tutors for questions during independent work time
- Our starting point: differing versions of course offered by 5 different instructors

CS 333: Introduction to Operating Systems

- A 10-week required course surveying fundamental topics of Operating Systems
- Textbook: Silberschatz, Galvin, Gagne, Operating Systems Essentials
- Lab: Required 3 hour weekly lab
- Students submit lab answers electronically before leaving the laboratory
- Lab machines are disconnected from the network during lab sessions
- Talking between students is allowed
- Lab instructor leads the lab and answers questions as students work independently

The Laboratory Exercises

- Linux Shell Program: Fork, exec, and wait as a simple introduction to concurrency
- Introduction to Pthreads: Students write producer/consumer code using basic pthreads interface, then add mutexes to protect a shared variable.
- POSIX semaphores
- POSIX read/write locks: explores POSIX pthreads read-write locks for synchronization. Students are asked to add synchronization to a working piece of code they did not write that implements lists. In particular, they must protect the code using pthreads read-write locks.
- Implementing read/write locks using semaphores: students use semaphores to implement read/write locking in a multi-threaded application they did not write

Key Findings

- The Closed Lab structure has lead to a measurable improvement in consistency of minimum skill set across all students
- Implementing the Closed Lab environment is challenging: switch students out of their regular home directories, provide the right number of instructors in the lab to keep n students working
- Students completing the synchronization lab report a much greater understanding of how and when to use synchronization
Many personal computers and workstations have two or four cores (that is, CPUs) that enable multiple threads to be executed simultaneously. Computers in the near future are expected to have significantly more cores. To take advantage of the hardware of today and tomorrow, you can parallelize your code to distribute work across multiple processors. Parallel Processing (TCP), National Science Foundation (NSF), and the sister communities, including ACM, has taken up proposing curriculum for computer science (CS) and computer engineering (CE) undergraduates on parallel and distributed computing. The goal of this committee has been to propose a core curriculum for CS/CE undergraduates, with the premise that every such undergraduate should achieve a specified skill level. The primary task identified was to propose a set of core topics in parallel and distributed computing for undergraduate curricula for CS and CE students. This liberal arts emphasis on broader concepts that are not tied to specific languages, architectures or operating systems is very much in line with the broader spirit of the recommendations stated in the NSF/TCPP Curriculum Initiative 2. According to LACS, a key component of any liberal arts perspective to computer science should include multiple problem-solving paradigms. However, they acknowledge the difficulty of doing this effectively within existing curriculum requirements. Our approach to this integration of the NSF/TCPP curriculum with the LACS Liberal Arts Curriculum requires no additional courses to be offered. Instead, it challenges instructors to rework existing courses to introduce these ideas often as a supplement to existing material. United States > Portland State University web ranking & review including accreditation, study areas, degree levels, tuition range, admission policy, facilities, services and official social media. Portland State University (PSU) offers courses and programs leading to officially recognized higher education degrees such as bachelor degrees, master degrees, doctorate degrees in several areas of study. See the uniRank degree levels and areas of study matrix below for further details. This 74 years old US higher-education institution has a selective admission policy based on entrance examinations and students' past academic record and grades. The admission rate range is 90-100% making this US higher education organization a least selective institution. Portland State University. Portland State national anthem duet went viral reunite for holiday song. This spring, Portland State will graduate its first class of Four Years Free students. The program, which pays tuition and fees for eligible undergrads, launched in 2017 and has enrolled 2,258 students. Video. Duo whose Portland State national anthem duet went viral reunite for holiday song.