EA: Spring-11: Expanding PDC Coverage in a Math-Focused CS1 and a (Functional) PL Course

Joe Kaylor • Konstantin Läufer • Chandra Sekharan • George K. Thiruvathukal
Department of Computer Science • Loyola University Chicago • USA
www.cs.luc.edu • laufer@cs.luc.edu • edupar.cs.luc.edu

Abstract/Current Focus
AY 2013-14: 1) Enhanced PDC coverage in a mathematically focused version of CS1 (CS 215) and 2) expanded PDC coverage in our programming languages courses (CS 372).

Prior Outcomes
Spring 2011: Three-week three-PDC course modules (20% of our 15-week semesters) targeting three required 2nd-year courses.
AY 2011-12: Four-week advanced PDC course modules in programming and distributed computing targeting electives typically offered every three semesters.
AY 2012-13: Moving PDC topics further down into CS1 and CS2, fleshing out PDC coverage in our intermediate object-oriented development course (CS 313), and stepping up evaluation.

Institutional Profile
• Urban, priv. Jesuit, lib. arts, ~16k students
• College of Arts and Sciences, ~8k students
• Dept. of Computer Science, ~300 students
• 10 FT faculty: 9 CS, 1 bioinf. (1/2 FTE), 1 algobirat (1/4 FTE)
• ~140 undergrad majors in CS, SE, IT, Networks & Security, bioinformatics
• ~50 US minors in CS, computer forensics
• ~110 master’s students in CS, SE, IT
• External funding: NSF S-STEM, NSF BPC lead institution, NSF research grants, industry

Early adopters: ~97 concurrency course, OOPSLA ’98 edu symm paper, HPBC book

Carnegie Classification
• Level: 4-year or above
• Control: Private not-for-profit
• Undergraduate Instructional Program: Bal/HGC
• Graduate: Comp/Doc*/Med/Vet
• Enrollment Profile: MU
• Undergraduate Profile: FT4/MS/HTI
• Size and Setting: L4/R
• Basic: RU/H: Research Universities (high research activity)
• Community Engagement: Curricular, Outreach, Partnerships
• CS/SE/IT: up to masters’ level

Where Our Graduates Go
• Industry ~75%
• midwest, coast, international
• consulting, finance, software, telecoms, ...
• Academia and Government ~ 15%
• Argonne, county admin, local universities
• Graduate School ~ 5%
• local, national
• Professional Schools ~ 5%
• business, law, medical

CS1
• Loyola course number: COMP 170
• includes some material on numerical methods at the K and C levels
• about 9 class hours were dedicated to sequential and parallel versions of these algorithms and the possible resulting speedup using data parallelism in C
• example: threads for speeding up trapezoidal rule integration

CS1 with mathematical focus
• Loyola course number: COMP/MATH 215
• Planned for Spring 2015
• includes some material on numerical methods at the K and C levels
• about 9 class hours were dedicated to sequential and parallel versions of these algorithms and the possible resulting speedup using data parallelism in C
• example: threads for speeding up trapezoidal rule integration

Intermediate Object-Oriented Development
• Loyola course number: COMP 313
• since fall 2011: CS, emphasis on PDC topics
• since fall 2012: Java with Android as highly effective context for studying concurrency and distributed computing topics (C and A levels)
• subject of forthcoming NSF/TCPP CDER PDC book chapter
• double 18-hour PDC module (intervened)
• concurrency and coordination (C)
• design forces: safety, liveness, performance (C)
• asynchronicity: external events and internal timers (A)
• Background threads (A)
• progress reporting and cancellation (A)
• offloading computation from mobile device to cloud (C/TA)
• throughput-latency tradeoff (C)
• Example: countdown timer
  • external events: button press
  • internal timers: one-shot and recurring
  • model-view-adapter (MVA) architecture

Programming Languages
• Loyola course number: COMP 372
• gradually increasing PDC coverage starting in spring 2010
• 10.5-hour PDC module on
  • mutable versus immutable state (C)
  • asynchrony/reactive programming, futures and promises (A)
  • threads (C)
  • actors (M)
  • other paradigms (F), e.g., software-transactional memory, task-parallel

Position Statements
[for discussion]
• To teach PDC topics effectively, they should not be taught in isolation. Instead, they should be taught in the context of relevant software engineering best practices.
• Mobile applications backed by RESTful cloud services are rapidly emerging as the mainstream paradigm for computing [Christensen, OOPSLA ’99]. PDC curricula should embrace it.

Eval: Learning Assessment
Sample concurrence questions: Suppose we have two philosophers. The first one, Kant, repeatedly believes the same thing for 10 minutes. Do you think Kant is a complete moron? The second one, Hubble, repeatedly believes the opposite thing for 10 minutes. Do you think Hubble is a complete moron?

Eval: Course Effectiveness
Sample questions (5pt Likert scale): Rate BEFORE/AFTER taking this course
• your event-programming expertise
• your thread-based concurrency expertise
• your cloud computing expertise
• your feeling of preparedness for the job market
• the effectiveness of Android for learning event-based programming
Results available at edupar.cs.luc.edu (Q3 code above).

Future Plans
Computer Systems/CS3
• Loyola course number: COMP 264
• offered every spring
• envisioning PDC module with suitable architecture, programming, and cross-cutting topics

Algorithms/CS7
• Loyola course number: COMP 363
• offered every fall
• developing PDC module: models of computation and complexity, basic algorithmic paradigms, and effective problem-solving frameworks and their algorithmic solutions

Evaluation
• Once our course modules have stabilized, we will need to measure their effectiveness longitudinally over a three- to five-year period.
• Building on the current evaluation process, we plan to work with Loyola’s Center for Science & Math Education, as well as the TCP and fellow early adopters.

Dissemination: workshops for subsequent adopters in the Midwest