Active Learning Materials for Computer Architecture and Organization

Brandon Myers
Computer Science
University of Iowa
- For a variety of active learning instruction strategies, there is evidence that they improve engagement in CS courses.
- Yet adoption is low.
Barriers to adoption of research based instruction strategies

National survey of 122 ECE faculty who had recently taught a sophomore engineering science course.

[Froyd et al, 2013]
Barriers to adoption of research based instruction strategies

Survey of 32 POGIL CS instructors, 25 from 4-year institutions

[Hu et al, 2016]
One such research-based instruction strategy: Process oriented guided inquiry learning (POGIL)

More on what it is later...

POGIL or POGIL-like instruction improved student performance in CS1 compared to using traditional instruction [Beck 2013, Hu 2014]
CS-POGIL: Process Oriented Guided Inquiry Learning in Computer Science

Activities by Course or Subject
(based on ACM/IEEE Computer Science Curriculum 2008)

<table>
<thead>
<tr>
<th>Course</th>
<th>Subject</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECS (Exploring Computer Science)</td>
<td>CSP (Computer Science Principles)</td>
</tr>
<tr>
<td>IntroCS (Introductory CS)</td>
<td>CS1 (1st course for majors)</td>
</tr>
<tr>
<td>AR. Architecture &amp; Organization</td>
<td>AL. Algorithms &amp; Computation</td>
</tr>
<tr>
<td>DS. Discrete Structures</td>
<td>GV. Graphics &amp; Visualization</td>
</tr>
<tr>
<td>IM. Information Management</td>
<td>IS. Intelligent Systems</td>
</tr>
<tr>
<td>PF. Programming Fundamentals</td>
<td>PL. Programming Languages</td>
</tr>
<tr>
<td>SE. Software Engineering</td>
<td>SP. Social &amp; Professional Issues</td>
</tr>
<tr>
<td>non-CS activities</td>
<td></td>
</tr>
</tbody>
</table>

http://cspogil.org/

200 POGIL activities
Tagged by maturity
Computer Organization and Architecture

Not enough POGIL activities to support the Computer Organization course
Project goal: write, pilot, revise, disseminate 6-8 POGIL activities to support the course
Computer Organization and Architecture

Project goal: write, pilot, revise, disseminate 6-8 POGIL activities to support the course

enough activities to teach a whole course and study effectiveness
Background

Computer Organization and POGIL

Example activity

Piloting

Research study
POGIL

process oriented teams of 3-4 students,
develop process skills

guided inquiry activities follow a learning cycle:
exploration, concept introduction, application

learning student centered
Aside: POGIL community

• Endorsed collections of activities
• Many 1-3 day training workshops each year
• Originated in physical sciences
• Has been adopted by Medical, Pharmacy, Family Science, Aviation, Library Science, and Computer Science
• Consistent SIGCSE presence, cspogil.org
Structure of a POGIL activity

- Exploration
- Application
- Concept Invention

The diagram shows the cyclical nature of a POGIL activity, moving from exploration to application, then to concept invention, and back to exploration.
Structure of a POGIL activity

Explore a *model* (plot, table, ...) by answering comprehension questions
Answer convergent questions leading to new concept

- **Exploration**
- **Application**
- **Concept Invention**
Structure of a POGIL activity

Exploration

Application

Concept Invention

Start to see patterns or generalize
Activity gives the discovered concept a term
Structure of a POGIL activity

Apply the concept in a new problem
Answer divergent questions to connect the concept to broader context
POGIL

Background

Computer Organization and POGIL

Example activity

Piloting

Research study
Computer Organization and POGIL

circuits, programs, documentation, waveforms make for good models

explore proposals for solving a problem (e.g., implementing procedure calls in assembly)
Computer Organization and POGIL

Exploration

Application

Concept Invention

Computer Organization is has a lot of concepts (two’s complement, ISA, compiler, assembler, datapath, control, addressable memory, stored program, heap, call stack, register file, microarchitecture, latency, throughput, ALU, ...)

Learn at the application level (translate this instruction to machine code) to prepare for the synthesis level (write an assembler).

Given a specification or problem, implement or solve it.
POGIL

Background

Computer Organization and POGIL

Example activity

Piloting

Research study
Example Activity
CS:2630 Computer Organization @ UI

• Required course for CS BS/BA majors
  • Pre-requisites CS2, Discrete Math
  • Required before: OS, Networking, Security
• The only numbered undergraduate course focusing on digital logic and computer architecture
• Recently 55-70 students/semester, ~30 in Summer
• Starting Fall 2018: 100 students between two sections
• Summer 2017 – present, taught in TILE active learning classrooms
TILE classrooms

• transform interact learn engage
• instructors supported by at least 6 hours of required training on student-centered pedagogy

https://teach.its.uiowa.edu

https://classrooms.uiowa.edu/van-350
Pilot

- Summer 2017 – 36 students, 8 weeks, 5 hr/week
- 6 draft POGIL activities, filling about 11 of 32 class meetings (1.25 hr)
- Self-formed informal teams, which mostly stayed constant
- 1 TA helped during these meetings
- Other meetings: lecture, think/pair/share, and software labs
Pilot outcomes: Authorship

- piloting reveals opportunities to elaborate or decompose exploration questions
- mark divergent questions to help teams manage time
Pilot outcomes: Facilitation

- tradeoffs between synchronous and asynchronous
- teams that are together a few weeks are more efficient and have the opportunity to improve process skills
- consider more frequent formative assessment and feedback
- incentivizing individuals becomes a challenge
- hold students accountable for before-class preparation
Curiosity...

What specific practices in an active learning pedagogy are effective in this intermediate systems courses?
Research questions

Compared to a mix of lecture, think/pair/share, and POGIL does solely POGIL with daily team feedback

• Improve learning outcomes?
• Improve self-efficacy in Computer Science?
• Improve self-efficacy in Computer Architecture?
• Help students with lower persistence improve their exam scores?
• Improve perceptions of peer and teacher interactions?
• Improve how students value Computer Architecture?
Study design

- Quasi-experimental study
- Pre/post test questionnaire
- Post test exams and concept inventory
- Fall 2017 – control (mix, no daily feedback)
- Spring 2018 – treatment (all POGIL, daily feedback)
- Fixed days of week, time, instructor, and classroom
- N = ~40 per semester
Aside: SOTL Umbrella IRB

Center for Teaching staff manage an ongoing IRB approved project

Streamlines the process for submitting an IRB application for a specific SOTL research project on campus

In CS terms: it is a well-designed interface

SOTL= Scholarship of Teaching and Learning
<table>
<thead>
<tr>
<th>Construct</th>
<th>Test</th>
<th>Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self efficacy (CS)</td>
<td>Pre/post</td>
<td>Self-efficacy excerpt of <em>Computer Science Attitudes (CSA)</em></td>
</tr>
<tr>
<td>Self-efficacy (architecture)</td>
<td>post</td>
<td>Inspired by <em>Computer Self-efficacy</em> and <em>Debugging Self-efficacy</em></td>
</tr>
<tr>
<td>Attribution</td>
<td>Post</td>
<td></td>
</tr>
<tr>
<td>Persistence</td>
<td>Pre</td>
<td><em>Grit scale</em></td>
</tr>
<tr>
<td>Peer and teacher interactions</td>
<td>Post</td>
<td><em>Student Personal Perception of Classroom Climate (SPPCC)</em></td>
</tr>
<tr>
<td>Engagement</td>
<td>Different</td>
<td>How much time?; <em>Student Engagement in Instructional Activity</em></td>
</tr>
<tr>
<td></td>
<td>pre/post</td>
<td></td>
</tr>
<tr>
<td>Task value</td>
<td>Pre expected/post actual</td>
<td>Course learning activities</td>
</tr>
<tr>
<td>Intrinsic value</td>
<td>Pre/post</td>
<td></td>
</tr>
<tr>
<td>Teamwork preference</td>
<td>Pre/post</td>
<td>Teamwork excerpt of <em>Work Style Inventory Quiz (Wilson)</em></td>
</tr>
</tbody>
</table>
Data to measure learning

- Exam scores
- Digital Logic Concept Inventory score (post test)
- Prior achievement and enrollment
No results yet!

Currently collecting data for the treatment semester
POGIL

Background

Computer Organization and POGIL

Example activity

Piloting

Research study
Collaborators

• Joint work with
  • Yeajin Ham, PhD student in Education Psychology @ UI

• Help with SOTL IRB from
  • Anna Flaming, Center for Teaching @ UI
  • Jane Russell, Center for Teaching @ UI

• Mentorship from Empirical CS Ed
  • Sarah Heckman, NCState
  • Mark Sherriff, UVA
  • Jeff Carver, U Alabama

• Mentorship on POGIL
  • Clif Kussmaul, Muhlenberg College
  • Renee Cole, ChemEd @ UI
Acknowledgements

• Graphics
  • experiment by Dinosoft Labs from the Noun Project
  • User Testing by Sharon Showalter from the Noun Project
POGIL

Background

Computer Organization and POGIL

Example activity

Piloting

Research study