Effective Closed Labs in Early CS Courses: Lessons from Eight Terms of Action Research

Elizabeth Patitsas and Steve Wolfman

University of British Columbia

SIGCSE 2012 (Raleigh NC)

March 6, 2012

We ask...

Think: what do you wish you knew about how your labs were going? [Write this down.]

What did you come up with?

This is what we wanted to know: why is perception of our labs so poor?

So we set out to do that, developing an approach based on action research.

Goal of the talk: convince you our approach is useful to solving your problem.

We will return to our opening question as a means of structure this talk.

So we have some labs...

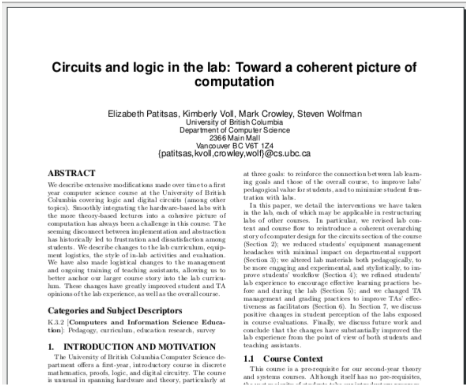
A focus group of students in 2005 identified the labs of our course as being unrewarding and unrelated to the lectures.

We were motivated to improve the labs – not only to make them more rewarding and relevant to students – but for the teaching assistants, too.

Perception is what we are concerned about here, not learning outcomes.

We made changes to the course. Things improved.

Then we wrote a WCCCE paper



Circuits and logic in the lab: Toward a coherent picture of computation

Elizabeth Patitsas, Kimberly Voll, Mark Crowley, Steven Wolfman
University of British Columbia
Department of Computer Science
2366 Main Mall
Vancouver BC V6T 1Z4
{patitsas,kvoll,crowley,wolf}@cs.ubc.ca

ABSTRACT

We describe extensive modifications made over time to a first year computer science course at the University of British Columbia covering logic and digital circuits (among other topics). Smoothly integrating the hardware-based labs with the more theory-based lectures into a cohesive picture of computation has always been a challenge in this course. The seeming disconnect between implementation and abstraction has historically led to frustration and dissatisfaction among students. We describe changes to the lab curriculum, equipment logistics, the style of in-lab activities and evaluation. We have also made logistical changes to the management and ongoing training of teaching assistants, allowing us to better anchor our larger course story into the lab curriculum. These changes have greatly improved student and TA opinions of the lab experience, as well as the overall course.

Categories and Subject Descriptors

K.3.2 [**Computers and Information Science Education**]: Pedagogy, curriculum, education research, survey

1. INTRODUCTION AND MOTIVATION

The University of British Columbia Computer Science department offers a first-year, introductory course in discrete mathematics, proofs, logic, and digital circuitry. The course is unusual in spanning hardware and theory, particularly at at three goals: to reinforce the connection between lab learning goals and those of the overall course, to improve labs' pedagogical value for students, and to minimize student frustration with labs.

In this paper, we detail the interventions we have taken in the lab, each of which may be applicable in restructuring labs of other courses. In particular, we revised lab content and course flow to reintroduce a coherent overarching story of computer design for the circuits section of the course (Section 2); we reduced students' equipment management headaches with minimal impact on departmental support (Section 3); we altered lab materials both pedagogically, to be more engaging and experimental, and stylistically, to improve students' workflow (Section 4); we refined students' lab experience to encourage effective learning practices before and during the lab (Section 5); and we changed TA management and grading practices to improve TAs' effectiveness as facilitators (Section 6). In Section 7, we discuss positive changes in student perception of the labs exposed in course evaluations. Finally, we discuss future work and conclude that the changes have substantially improved the lab experience from the point of view of both students and teaching assistants.

1.1 Course Context

This course is a pre-requisite for our second-year theory and systems courses. Although itself has no pre-requisites,

As we wrote our experience report we realized we wanted more data to assess the changes we had made.

And data to identify where more change could happen.

Before we tackle methods...

We have a big course with about 130 students a term and about 12 TAs (mix of undergrad and grad). Statistical significance!

But unfortunately there's a large turnover in who teaches the course, both for faculty and teaching assistants.

Before we tackle methods...

We do have a positive teaching culture, and are able to engage the TAs in weekly lab meetings where they go over the labs in advance.

Those meetings matter since our TAs often have little background in digital logic.

The meetings also give the TAs social support and a sense of community in the course.

Action research!

Action research differs from more traditional approaches to research (e.g. positivism and constructivism). Our approach is based on action research.

Goal: to enact *social change*. We used action research to identify problems in the labs, and to assess changes intended to improve them.

Process: through empowering the stakeholders through participatory research.

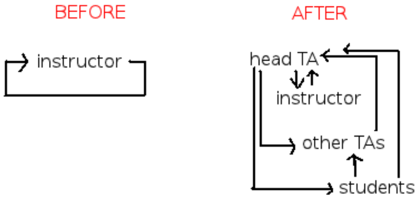
Role of the researcher: the researcher is a participant in the course; here we act simultaneously as course designers, assessors, and educators.

Returning to your thoughts

Remember: what do you wish you knew about how your labs were going?

Design process

Our adoption of an action research-based approach has changed the



We now involve all stakeholders in the design of the labs

informally soliciting feedback from staff meetings and labs

formally soliciting their feedback through surveys and diary-like "lab documents".

We measure perception, as this is our project's goal.

CPSC 121 Lab Feedback

We are in the process of redeveloping many of the labs in this course to improve them. For one mark in lab each lab, please provide feedback on how you found the lab. Feedback is anonymous - after hitting submit, show the ensuing page to your TA for the mark.

* Required

What is the number of the lab you are reviewing? *

I had enough time to work on this lab.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

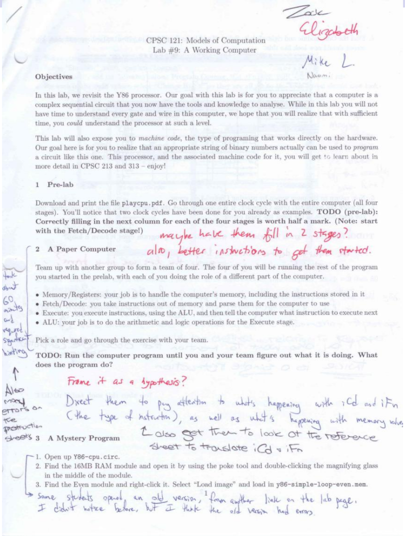
The written instructions were clear and well-written.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The lab was relevant to the lecture material.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The lab was interesting.



Zack
Elizabeth
Mike L.
Naomi

CPSC 121: Models of Computation
Lab #9: A Working Computer

Objectives

In this lab, we revisit the Y86 processor. Our goal with this lab is for you to appreciate that a computer is a complex sequential circuit that you now have the tools and knowledge to analyse. While in this lab you will not have time to understand every gate and wire in this computer, we hope that you will realize that with sufficient time, you *could* understand the processor at such a level.

This lab will also expose you to *machine code*, the type of programing that works directly on the hardware. Our goal here is for you to realize that an appropriate string of binary numbers actually can be used to *program* a circuit like this one. This processor, and the associated machine code for it, you will get to learn about in more detail in CPSC 213 and 313 – enjoy!

1 Pre-lab

Download and print the file `playcpu.pdf`. Go through one entire clock cycle with the entire computer (all four stages). You'll notice that two clock cycles have been done for you already as examples. **TODO (pre-lab): Correctly filling in the next column for each of the four stages is worth half a mark. (Note: start with the Fetch/Decode stage!)**

maybe have them fill in 2 stages?
also, better instructions to get them started.

2 A Paper Computer

Team up with another group to form a team of four. The four of you will be running the rest of the program you started in the prelab, with each of you doing the role of a different part of the computer.

- Memory/Registers: your job is to handle the computer's memory, including the instructions stored in it
- Fetch/Decode: you take instructions out of memory and parse them for the computer to use
- Execute: you execute instructions, using the ALU, and then tell the computer what instruction to execute next
- ALU: your job is to do the arithmetic and logic operations for the Execute stage.

Pick a role and go through the exercise with your team.

took about 60 minutes and required significant hinting

TODO: Run the computer program until you and your team figure out what it is doing. What does the program do?

Frame it as a hypothesis?

Direct them to pay attention to what's happening with iCd and iFn (the type of instruction), as well as what's happening with memory values
↳ also get them to look at the reference sheet to translate iCd & iFn

Also many errors on the instruction sheets

3 A Mystery Program

1. Open up `Y86-cpu.circ`.
2. Find the 16MB RAM module and open it by using the poke tool and double-clicking the magnifying glass in the middle of the module.
3. Find the Even module and right-click it. Select "Load image" and load in `y86-simple-loop-even.mem`.

Some students opened an old version, from another link on the lab page. I didn't notice before, but I think the old version had errors.

1

CPSC 121 End of Term Lab Survey

As part of our endeavour to improve the CPSC 121 labs, we would like your feedback on how the labs were overall this term. Filling the survey is worth one bonus mark in lab 10. Your answers, as usual, are confidential.

The labs contributed to my understanding of the course material.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The pre-labs contributed to my understanding of the course material.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The challenge problems contributed to my understanding of the course material.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Feel free to comment on any of the above questions:

Elizabeth Patitsas and Steve Wolfman

CPSC 121 Lab TA feedback

As part of our curriculum development initiative, we'd like to know how you (the TAs) found the 121 labs.

The labs were interesting for me

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The labs were fun for me

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

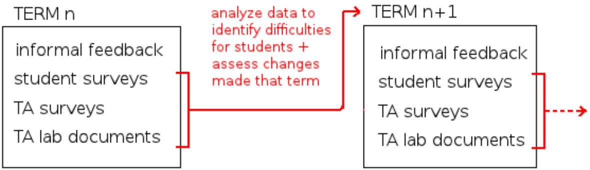
The labs were rewarding for me

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

The labs were of an appropriate difficulty

	1	2	3	4	5

The Process



We solicit qualitative and quantitative data from the students and TAs are analyzed at the end of every term, to assess how the labs went that term.

Changes in term n are assessed versus term $n-1$.

We identify the source of greatest pain for the students in term n and make changes in term $n+1$ to try to counteract it.

Returning to your thoughts

Remember: what do you wish you knew about how your labs were going?

Example results

An example of change

In our first end of term student surveys
: both qualitative and quantitative data agreed: "*tkgate sux!!!!!* [sic]". (We also learnt that the CPU lab was poorly received, and that the students loved the regular expressions lab.)

In the next two terms
: we tried improving our support of tkgate. Still no change in student feedback. "*tkgate is totally not user friendly.*"

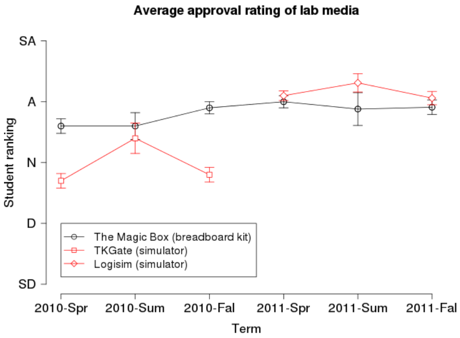
So we threw it out.
: We switched to logisim.

And then what happened?

Qualitative feedback that term, and subsequent ones, didn't have complaints about the circuit simulator.

Quantitative feedback, however, jumped up. Dramatically.

And note that feedback on breadboarding has held steady without any major changes to it.



How about another example?

The TAs provided a lot of useful feedback within our community.

They complained that the sequential circuitry lab had too many new concepts in it: clocking, latches, DFFs, sequential circuitry, counters.

So we broke it up. Clocking was moved to lab 1. We added another week on sequential circuitry.

TA feedback on these labs improved. (Students were still lukewarm to sequential circuitry.)

On the note of breaking labs up

Elizabeth Patitsas and Steve Wolfman

The weekly end of lab surveys allowed us to realize students liked labs more when *they could complete them on time*.

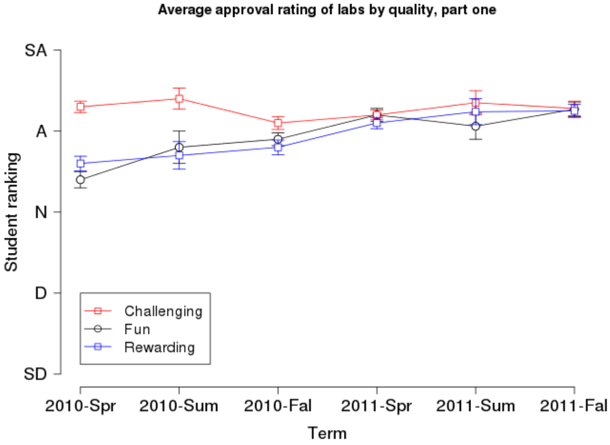
So we shortened our labs and when we saw the rating of "I could complete the lab in the allotted time" improve...

We also found they were rated as more fun, rewarding and interesting.

To shorten the labs we identified the *goals* for every lab and *cut everything else out*.

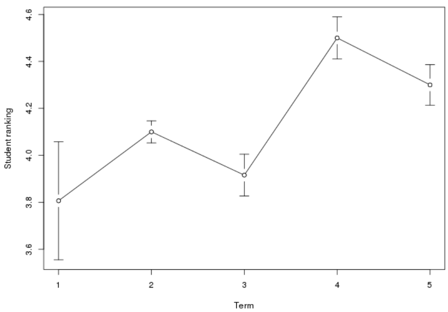
Trends over time

We'd like for everything to be increasing gradually like this:



Sometimes, things don't always get better

Average student ranking of lab 3 (had time to complete)



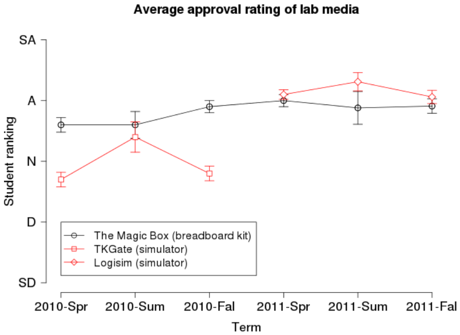
While our approach allows us to identify problem areas in the labs, it doesn't tell us the solution path.

We experiment by making changes through the community suggestions.

Discussion

Sustaining change

Not only have we produced measurable improvements in perception, we see the improvements last for subsequent terms.



Sustaining our change

The our improvements have lasted through five head TAs and seven course instructors.

So if you're wondering about the Hawthorne effect – we've seen improvements in terms where we weren't on the course – and the students change term to term.

Changing our community

A result of action research: change in the teaching community of this class. This is a deeper change than the changes we have made to the labs themselves.

We have empowered the TAs, who now freely give their input on the labs – sometimes pushing back on the instructors.

Factors for lasting change

And we know from the education literature that community approaches have more staying power.

Indeed, "proceeding with a non-threatening, incremental pace of change;" and "mutual trust amongst stake-holders" were listed as conditions that promote and sustain changes in curriculum change (Jones, 2002).

Jones. 2002. Transforming the Curriculum: Preparing Students for a Changing World. ASHE-ERIC Higher Education Report. Jossey-Bass Higher and Adult Education Series.

Next steps

We measure perception, and have improved it in a sustainable manner. We now want to look at student learning.

You may care about things other than perception.

Think: what do you wish you knew about how your labs were going?

Then: how could you find that out?

Conclusions

For our course we used the action research-based approach we just described.

Our goal of improving the perception of the labs has been satisfied.

The process of empowering our teaching community to improve the labs has produced lasting change.

And we hope that you can adapt our approach to improving your labs.

Thank yous

Head TAs over the years: Rachel Busby, Mark Crowley, Ian Dewancker and Vanessa Kroeker

Instructors over the years: Meghan Allen, Patrice Belleville, Dave Tompkins, George Tsiknis, Kim Voll and Bob Woodham

The many TAs over the years (they won't all fit on here!)

And the many students over the years who gave us their feedback!

Plus Michele Ng for the focus groups, Anthony Winstanley for lab support.

And support from CSSEI, NSERC.

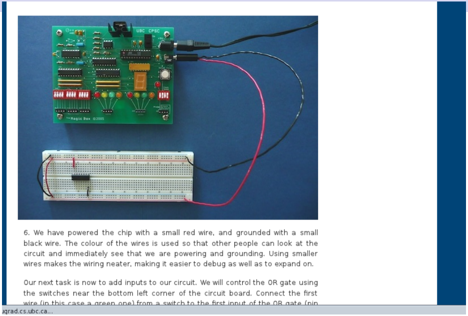
The end of lab surveys

The end of lab surveys gave us feedback as the term progressed – not just what the students remembered at the end.

Around labs 2 and 3, students started complaining that "*I still don't get how to start using the magic box... wish we'd an actual introduction to it!*"

The TAs independently concurred that their students needed more guidance from the start on how to do breadboarding.

And then what happened?



We wrote a new lab 1 with a picture-based walkthrough on how to do breadboarding, based on this community process.

The TAs who were on the course both terms noted their students handled breadboarding better thereafter.

The students had noticeably fewer complaints about breadboarding in subsequent terms. They rated breadboarding labs as more fun!