Extending and Improving the Role of Deliberate Practice in CPSC 259

Edwin M. Knorr
Department of Computer Science

CWSEI End-of-Year Event: April 11, 2014

Background

- CPSC 259 is a core computing course for Electrical Engineering students who are *not* in the Computer Engineering option.
- CPSC 259 is a course in:
  - Intermediate C Programming
  - Data Structures
  - Basic Algorithms, including Big-O complexity analysis and a small amount of discrete math
- Enrolment (one section annually):
  - 133 students in April 2012
  - 197 students in April 2013
  - 191 students in April 2014

Grading Structure: 2013 vs. 2014

2013: 40% for term work, 60% for exams

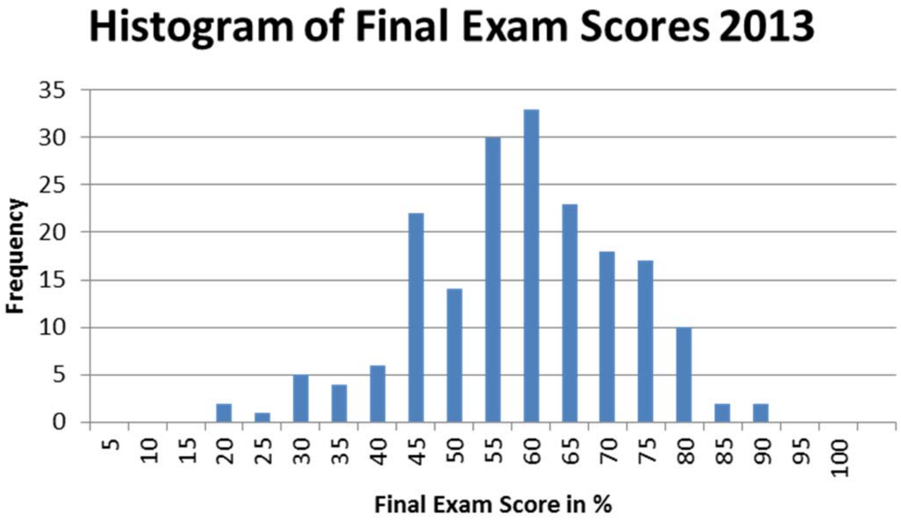
- 3% for Connect quiz questions (6 sets of 5-6 multiple-choice questions, for participation marks only)
- 3% for in-class clicker questions—half for participation, half for correctness; best 90% of attempts
- 10% for in-lab participation, pair programming, and deliverables (5 assignments)
- 24% for take-home programming assignments (5)
- 18% for the midterm
- 42% for the final exam

2014: 32% for term work, 68% for exams—assuming in-lab tests are "exams"

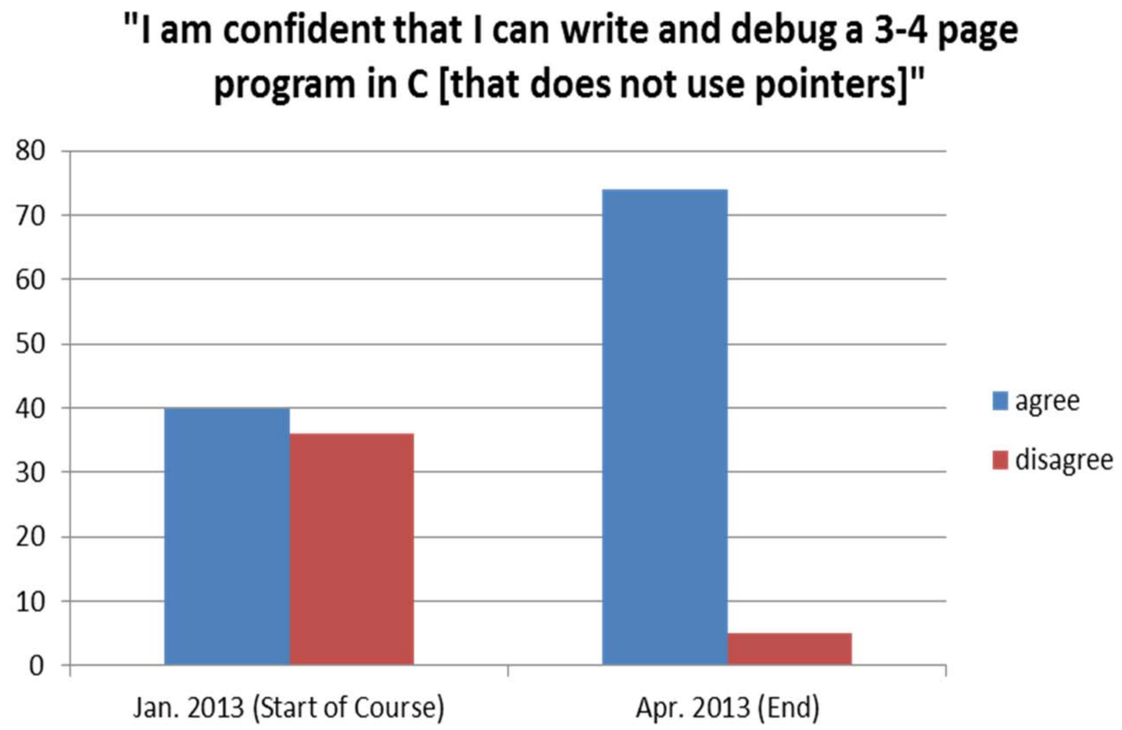
- 4% for Connect quizzes (9 sets of 5-8 multiple-choice questions that students can take as many times as they like; but, we'll only count their last attempt)
- 4% for in-class clicker questions—half participation, half correctness; best 90%
- 9% for in-lab participation, teamwork, and deliverables (5 assignments)
- 15% for take-home programming assignments (5)
- 10% for 5 in-lab programming tests (2% each)—approx. one hour each
- 18% for the midterm
- 40% for the final exam

Problems in 2013 and 2012

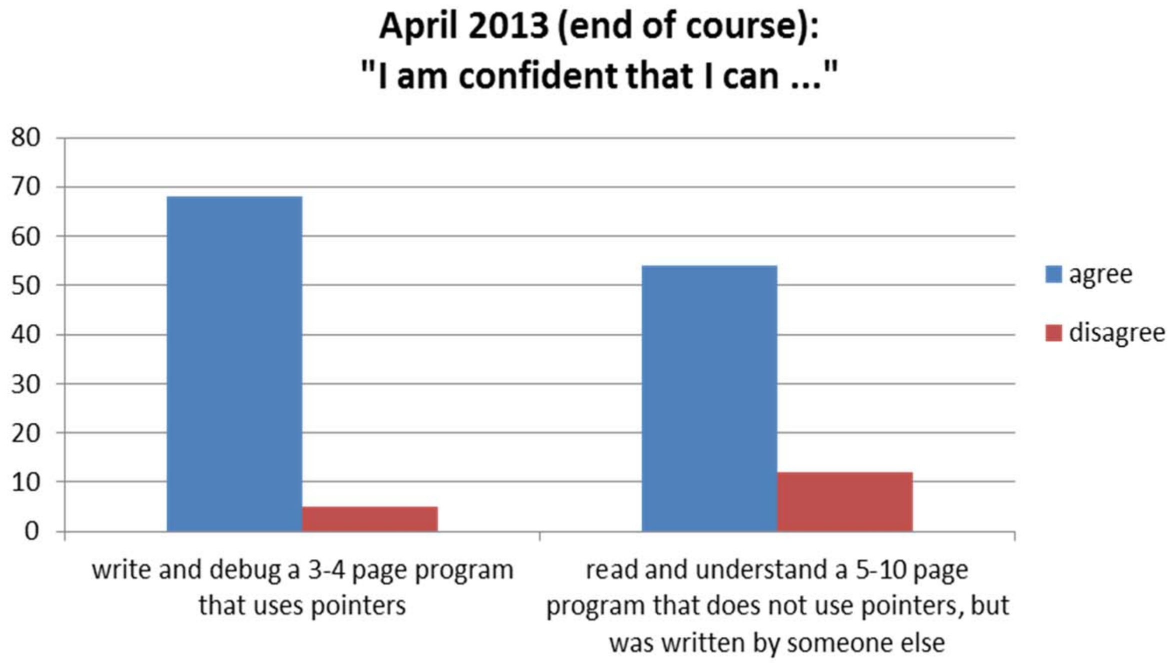
1. Poor performance on the midterms' programming questions
2. Poor performance on the final exams' programming questions
3. Relatively low confidence levels for programming



Feedback from 2013 Surveys Regarding Confidence Levels



Feedback from 2013 Surveys Regarding Confidence Levels

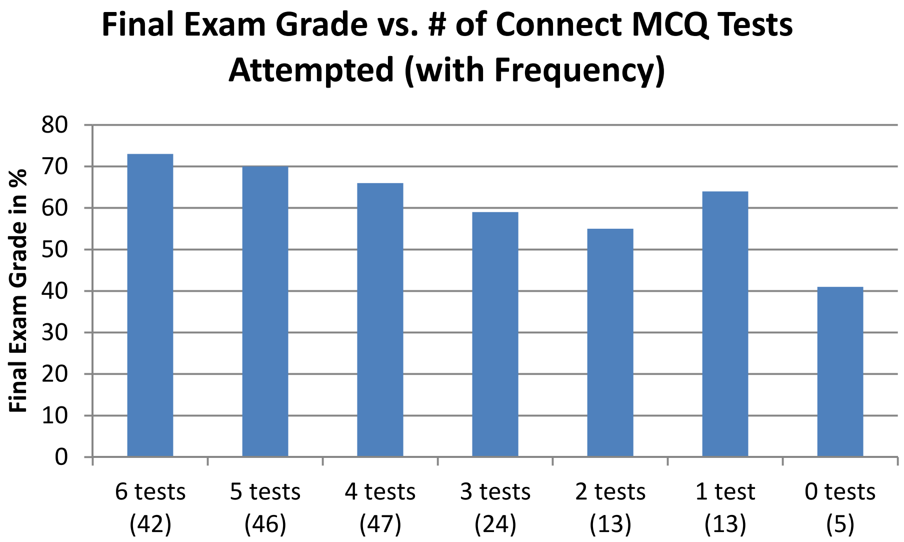


- This is still low, especially since this is their 2nd course in C.
- **GOAL**: Improve these confidence levels (in addition to improving their grades/learning).

What We're Doing to Rectify the Situation

2013 Interventions:

1. 6 multiple-choice tests on Connect for practice during the term (for a small number of participation marks)
2. New lab material on debugging and debuggers
3. An online, interactive tutorial/simulation on data types, memory, addressing, and pointers.



What We're Doing to Rectify the Situation (cont.)

2014 Interventions:

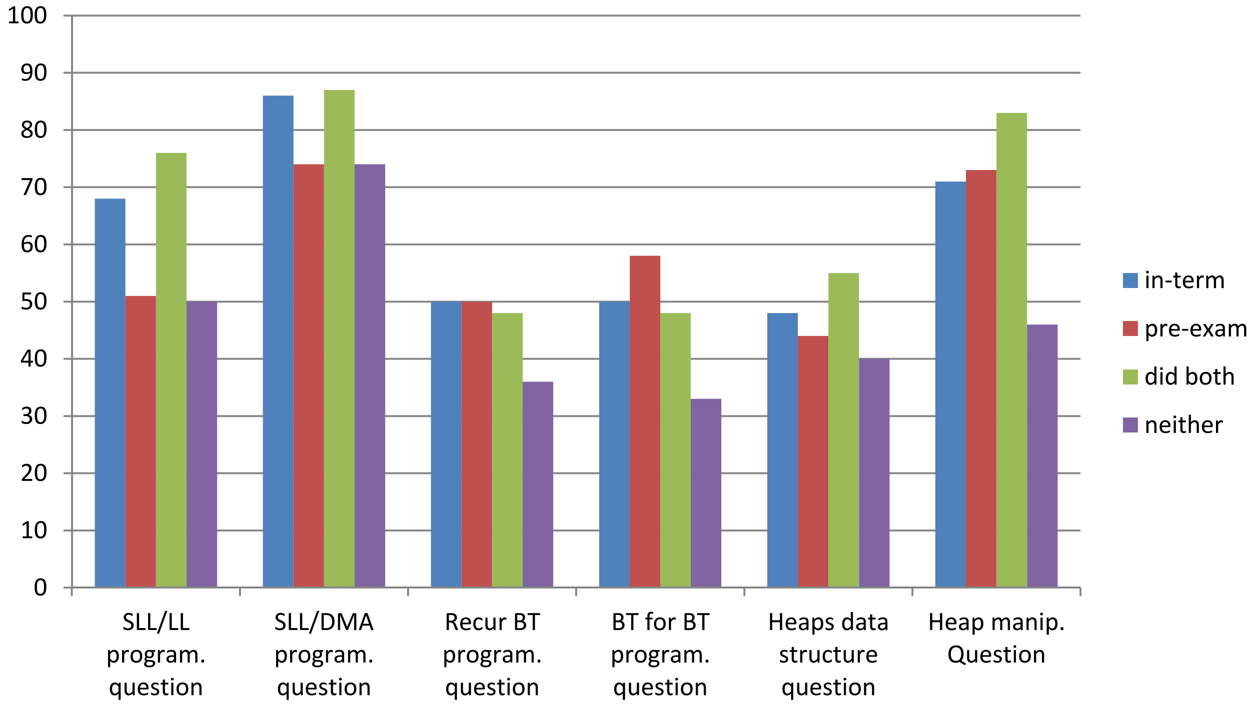
1. We deployed 3 more multiple-choice tests on Connect for in-term practice (for *correctness* marks this time, and not just for participation … again, allowing free re-takes).
2. We introduced 5 bi-weekly, in-lab, programming tests for 6 lab sections (30 new tests in all).
3. We provided and regularly promoted the free, Stanford, C programming tutorial guides.

To Do:

1. Compare the results/usage of Connect tests with each unit's corresponding final exam questions.
2. Correlate the in-lab programming results with this term's final exam question results.
3. Survey students' end-of-term confidence levels.

2013 Connect MCQ Tests: **Spaced repetition in testing.** (1) In-Term = Students who did the test *only* during the term for marks (just after the lecture unit); (2) Pre-Exam = Students who did the test *only* when it was re-released for practice (not for marks) before the final exam; (3) = Students who did BOTH (1) and (2); and (4) = Students who did NEITHER (1) nor (2).

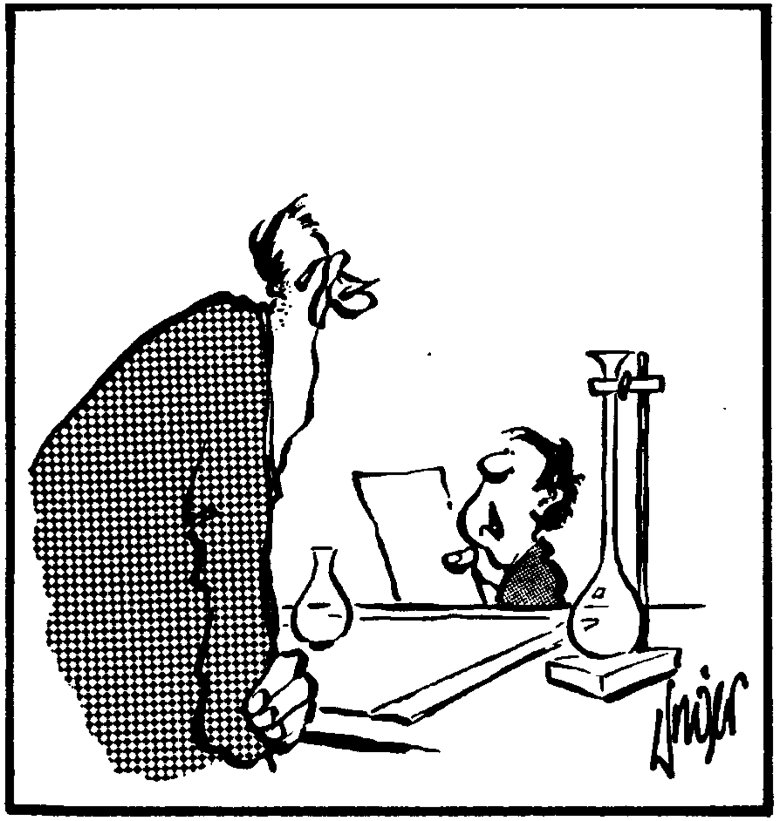
Repetition in Testing: Performance on Major Final Exam Questions for Categories 1-4



Why Introduce In-Lab Programming Tests?

- Students use the same compiler and computer environment that they write their programs on
  - Computer gives instant feedback (success, warnings, errors)
  - Students iteratively: read existing code, write code, compile, test, and debug
- "Real-world" conditions vs. paper-and-pen on an exam
- Provides practice at spaced intervals (bi-weekly)
- Keeps students accountable and engaged
- They become more effective contributors during pair programming

Summary: 4 Kinds of Tests in Jan-Apr 2014: 9 Connect MCQ Tests (Low Stakes), 5 In-Lab Programming Tests, Midterm, Final Exam



"I think you'll find my test results are a pretty good indication of your abilities as a teacher."

Confounding Factors

- Impact of one extra lecture hour per week
- Impact of officially changing the course from 2 credits to 4 credits
    - Realistically, it went from 3 credits to 4 credits
    - Some new course content
        - New material won’t be compared in our analysis
        - Most content remains the same as 2012
    - Allows us to inject bi-weekly programming tests into the lab environment