3-2009

It Seemed Like a Good Idea at the Time

James Huggins

Jonas Boustedt

Robert McCartney

Katherine Deibel

Beth Simon

See next page for additional authors

Authors

James Huggins, Jonas Boustedt, Robert McCartney, Katherine Deibel, Beth Simon, and Suzanne Westbrook

It Seemed Like a Good Idea at the Time

Jonas Boustedt
Dept. of Mathematics, Natural and Computer Science
Högskolan i Gävle
S-80176 Gävle, Sweden
jbt@hig.se

Robert McCartney (moderator)
Dept. of Computer Science and Engineering
Univ. of Connecticut
Storrs, CT 06269 USA
robert@cse.uconn.edu

Katherine Deibel
Dept. of Computer Science and Engineering
Univ. of Washington
Seattle, WA 98195-2350 USA
deibel@u.washington.edu

Jim Huggins
Dept. of Computer Science
Kettering University
Flint, MI 48504 USA
jhuggins@kettering.edu

Beth Simon
Dept. of Computer Science and Engineering
Univ. of California San Diego
La Jolla, CA 92093 USA
bsimon@cs.ucsd.edu

Suzanne Westbrook
Dept. of Computer Science
Univ. of Arizona
Tucson, AZ 85721 USA
sw@cs.arizona.edu

The Mystery Presenter

???

Categories and Subject Descriptors

K.3.2 [**Computer and Information Science Education**]: self-assessment; K.4.2 [**Social Issues**]: employment

General Terms

Design, Experimentation, Human Factors, Measurement, Performance, Verification

Keywords

fiasco, disaster, breakdown, failure, humiliation, termination

SUMMARY

We often learn of successful pedagogical experiments, but we seldom hear of the the ones that failed. For this special session we solicited submissions from the SIGCSE membership, selected the best from among these, and will have presentations at the session by the selected authors. Our contributions describe pedagogical approaches that seemed to be good ideas but turned out as failures. At the session, contributors will describe their pedagogical experiment, the rationale for the experiment, evidence of failure, and lessons learned.

1. OVERVIEW

An underlying assumption in the SIGCSE literature is that every decision we make in our class offerings is a small-scale local experiment to determine which acts can best lead to learning. In our community, as in most academic disciplines, there is an exclusive focus on the success of these experiments. Rarely, if ever, do we hear about the experiments that failed. Given that negative results can be as valid as positive results in the scientific endeavor, it seems natural that there should be a forum for the discussion of especially negative experiences in CS Education.

This is that session: a place for us to discuss the failures we have all had, with the hope that others can avoid the paths that led to these unanticipated results. This session is structured like Parlante's "Nifty Assignments" sessions: a set of presenters selected from submissions solicited by a general CFP to the SIGCSE community. Similar sessions at SIGCSE 2007 [2] and SIGCSE 2008 [1] were well-attended and wildly successful. See http://depts.washington.edu/goodidea/ for more details.

2. OVERALL OBJECTIVE

Much like Tolstoy's happy families, successful experiments are all similar but failed experiments all fail in their own way. However, not all failures are interesting or would make for a good discussion. We selected teaching interventions for which there was reasonable expectation of success, for example those based on previously published work. We also selected for reports that provided evidence of failure – or at least significant deviations from expected outcomes, reasons for the failure, and implications for practice for other CS teachers.

By presenting these experiences, the authors provide cautionary tales (and some entertainment) to other instructors. By providing a forum for the failed experiment, we hope to encourage risk taking in the classroom, and by focusing on evidence we hope to foster a community with a greater eye for documenting our classroom experiments.

SIGCSE'09, March 3–7, 2009, Chattanooga, Tennessee, USA.
ACM 978-1-60558-183-5/09/03.

Extra credit gone wrong (Katherine Deibel)

The tale of an assignment that leads to students unintentionally crashing a department's servers is nothing new. However, for this one assignment in a data structures course, the story is far form ordinary. The assignment involved using BST, AVL, and splay trees for calculating word frequencies, and had never caused a problem. One of the extra credit problems involved doing a performance analysis of the three trees with the Unix dictionary as input. In previous terms, only a few students ever did that extra credit task, but this time, most of the class decided to do the extra credit one hour before the assignment was due. Their programs quickly overwhelmed and crashed the undergraduate servers. Although the poor choice of an input file contributed to this disaster, the ultimate cause was a change to the grading practices that despite providing better feedback, pushed the students to gather extra points whenever possible.

A StoogeSort-inspired Exam Question (Jim Huggins)

In Summer 2008, I was searching for an exam problem for a take-home midterm in an algorithms course, which would ask students to analyze an algorithm for best-case and worst-case running times. The algorithm should be simple to understand and analyze, but not be commonly available via web searches.

Inspired by the classic StoogeSort, I invented a simple, eight-line, recursive sorting algorithm. A simple analysis showed that the algorithm had running time of $O(n)$ in the best case, and $O(2^n)$ in the worst case.

Students on the exam uniformly answered that the algorithm was not exponential, but polynomial in the worst case. Several students gave recurrence relations which were hideously complicated.

As I was preparing to hand back the exam, and berate my students for their failures, it occurred to me that the students might be right after all ...

Radically better students in 15 minutes (Beth Simon)

We've all experienced it – especially in CS1. Some students find one programming error – throw their hands up and either A) quit or B) ask you to debug their program for them. Others see an error, say "hmmm...", put in print statements, change a line of code, look at error messages, whatever. They make forward progress. One lone bug doesn't stand in their way. They *learn* from mistakes and, moreover, don't think that every bug is a something to be hated, to pound the desk about, to complain about... Why is this? How can we get the first set of students to behave like the second? Great news – the answer may lie in the work of Stanford psychologist Carol Dweck. According to her studies, we need to get students to adopt a growth mindset. We tried out a modification of one of her approaches. Well, it seemed like a good idea at the time...

This project was too big for just one person. Sharing the "credit" for these results are Brian Hanks, Laurie Murphy, Lynda Thomas, and Carol Zander.

Text Processing, or ParagraphFun (Suzanne Westbrook)

It is challenging to introduce students to object-oriented programming concepts while still providing good opportunities for problem-solving. In Spring 2000, I gave a text processing assignment from Cay Horstmann's Computing Concepts book that required determining multiple counts from a text, parsing words into syllables, and deriving an index for the author's writing level. The assignment seemed too challenging for most students, so I revised it by removing syllable identification and adding other things including counting palindromes, word substitution and, for fun, conversion to Pig Latin.

Students continued to have trouble and no one thought Pig Latin was as much fun as I did. I kept tinkering with the assignment each semester, making it easier and trying to give students more insight on how to approach it. They still hated it and often didn't even attempt it.

I finally gave up in defeat after assigning it *(number withheld)* times!

Title witheld to maintain anonymity (The Mystery Presenter)

Note: in the calls for the two previous "It Seemed Like a Good Idea at the Time" sessions, we offered presenters the option of making their presentation anonymously, in disguise. No one wanted to do it. However, this time we have a mystery presenter. Whether he or she uses the disguise is still under consideration, but the topic and description must remain secret for now.

3. REFERENCES

[1] J. Boustedt, R. McCartney, J. Tenenberg, S. D. Anderson, C. M. Eastman, D. D. Garcia, P. V. Gestwicki, and M. S. Menzin. It seemed like a good idea at the time. In *SIGCSE '08: Proceedings of the 39th SIGCSE technical symposium on Computer science education*, pages 528–529. ACM Press, 2008.

[2] J. Boustedt, R. McCartney, J. Tenenberg, T. Winters, S. Edwards, B. B. Morrison, D. R. Musicant, I. Utting, and C. Zander. It seemed like a good idea at the time. In *SIGCSE '07: Proceedings of the 38th SIGCSE technical symposium on Computer science education*, pages 346–347. ACM Press, 2007.