The Code Mangler Project

NICK CHENG AND BRIAN HARRINGTON
University of Toronto Scarborough
Toronto Ontario, M1C 1A8

ABSTRACT

One of the many issues facing someone who teaches introductory computer science is the amount of time (and hence resource) it takes to grade programming questions on tests. When asked to write code, students create a wide variety of solutions. The grader is usually given a sample solution. Sometimes a student solution differs greatly from the sample solution. In such circumstances it can often be nontrivial determining whether the student's code is correct, or how close it is to being correct. This impedes the grading process.
The **Code Mangler** is an attempt to reduce the variation in students answers, and correspondingly the effort required to mark those answers without compromising the integrity of the test. The idea is that by providing students "mangled" versions of the code, they are guided towards the desired solution, but in such a way that the question still accurately assesses their understanding of the material.

Methods

General approach to posing questions

The common way of posing a programming question is to give the specification for the code (i.e., describe what the code is supposed to do), then ask students to write the code. In devising new ways to pose such questions, our general approach was to start with the answer (the expected solution). We considered various ways to create questions for which the answer is the solution code, and we found two that seemed promising. So we tried them on the fall midterm test to gain some experience. Then, feeling confident that these types of questions are an improvement over the common type, we set out to prove it on the final exam.

Code Mangler question

A Code Mangler question gives the specification for the code. It also gives the lines of code from our solution, but in a mangled form. What we did specifically was to remove all comments and indentation, and shuffle the lines by sorting them. We used a narrative like this.

```
  Nick wrote a program that [description of what program does].
Then the CODE MANGLER struck!  He (she?)  [description of how
code was mangled].  Please help Nick to reconstruct his program.
```
[a]

Variations

There are some optional variations to Mangler questions. We would say so if any of these were used.

Removal of duplicate lines:

If multiple lines of code became the same (after removal of indentation), then there is an option to remove duplicate copies.

Introduction of monkey wrenches:

There is an option to include, in the mangled code, extraneous lines of code that are not part of the solution. We may add any line of code that corresponds to some typical student error. Alternatively we may just add extra lines to better hide the solution. Here are some examples.
If we have a line of code like $if(x == y)$: then we would add a line of code like $if(x! = y)$:
In our midterm test we had two Mangler questions, and we combined the mangled code for both questions.

Inclusion of some or all comments:

We may leave in the header comments (docstring), unmangled. We may also leave in the internal comments, but mangled with other lines of code.
Using these variations allows us to adjust the difficulty level of a question. Removing duplicate lines and introducing monkey wrenches makes a question harder, whereas leaving in comments makes it easier.

Code Stealer question

A Code Stealer question gives the specification for the code (same as a Mangler question). It also gives all the header and internal comments for the code, keeping the lines in order with their indentation. It is as if the Code Stealer took our solution and whited out every line of executable code. Essentially students are asked to fill in the blanks according to what the comments say. We used a similar narrative as with Mangler questions, saying that the Stealer is a distant relative of the Mangler.

[a] Aside: We had some students on the course discussion forum cursing Nick for not backing up his work!

The investigators would like to thank Sotirios Damouras for his help and advice on the statistical analysis of this experiment, as well as the students and TAs for CSCA48: Introduction to Computer Science II, Winter 2014 for being our test subjects and helping to foil the `Code Mangler` in his attempts to ruin our exams.

Experiment

We wanted to prove (or at least provide strong evidence) that:

1. Mangler questions take less time to grade than common type questions, and
2. Mangler questions test student ability to code as well as common type questions.

We also wanted to prove the same points for Stealer questions. However, our experience from the midterm test already left no doubt about point I. So with regard to Stealer questions, we designed our final exam with only point II in mind.
We created two versions, called version A and version B, of the final exam. There are 7 questions, with the first 5 being the same on both versions.

- Questions 1-4 are each graded out of 5, and questions 5-7 are each graded out of 10.
- Questions 1-3 are tracing questions, where students are given some code and asked what output it produces.
- Question 4 is about testing. It asks students to devise test cases.
- Question 5 is a Stealer question (about clubs).
- For questions 6 and 7, we took two programming questions and pose each one in two different ways, thus yielding 4 questions. Then we put the first question posed the first way and the second question posed the second way on version A, and the second question posed the first way and the first question posed the second way on version B. Here are the details.
  - Question 6 is a common type question. On version A, it is about cards. On version B, it is about grades.
  - Question 7 is a Mangler question, with monkey wrenches. On version A, it is about grades. On version B, it is about cards.
  - So each version has both a cards question and a grades question, and also both a common type question and a Mangler question.
  - On both versions the common type question comes before the Mangler question.
  - We consider the cards question to be more challenging than the grades question.

The final exam was written in a large room (the Gym). We alternated columns of seats with different versions of the exam. Students were not told, and likely not aware, that there were two versions. When all the exams were written and collected, we had 469 papers, with roughly equal numbers of version A and version B. Before grading questions 6 and 7, we randomly set aside 100 papers. When everything else was graded, we selected two groups of 4 TAs each.

- We gave 50 papers to each group.
  - We timed how long it group X to grade 50 grades questions posed as a Mangler question.
  - We timed how long it group Y to grade 50 grades questions posed as a common type question.
- Then the two groups swapped papers.
  - We timed how long it group X to grade 50 cards questions posed as a common type question.
  - We timed how long it group Y to grade 50 cards questions posed as a Mangler question.

For each student we recorded the marks they earned on every question as well as which version of the exam they wrote. Any necessary adjustments on total exam marks were made to ensure that students were unfairly disadvantaged by writing one version or the other. Before analyzing the data, all student identification was removed so to maintain anonymity and confidentiality.

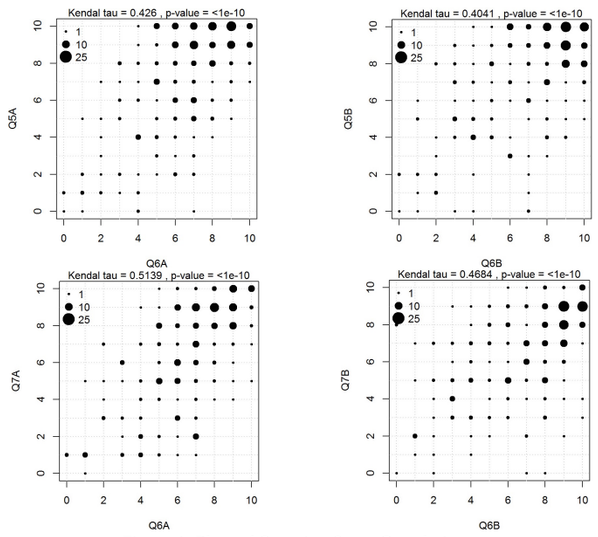

Figure 1: Plots of Question Score Correlations

Results

Grading 50 questions

Here are the times taken for the two groups of TAs to grade the two questions.

- Time for group X to grade 50 grades questions posed as a Mangler question: 9 minutes.
- Time for group Y to grade 50 grades questions posed as a common type question: 16 minutes.
- Time for group X to grade 50 cards questions posed as a common type question: 22 minutes.
- Time for group Y to grade 50 cards questions posed as a Mangler question: 20 minutes.

A poll of the marking TAs found that they felt the mangled versions of the questions were much easier to mark.
While posing a question as a Mangler question improved marking time significantly for the grades question, its effect was positive but minimal for the cards question. It seems the cards question is not only harder to answer for students, but also harder to grade for TAs. So for harder questions, mangling becomes less helpful.

Marks correlation

To see if Stealer and Mangler questions test student ability to code as well as common type questions, we generated the scatter plot graphs in Figure1 comparing marks earned by students from:

- the Stealer question (Q5A) versus the cards question posed as a traditional question (Q6A),
- the Stealer question (Q5B) versus the grades question posed as a traditional question (Q6B),
- the grades question poased as a Mangler question (Q7A) versus the cards question posed as a traditional question (Q6A),
- the cards question poased as a Mangler question (Q7B) versus the grades question posed as a traditional question (Q6B).

Larger dots in our graphs indicate higher number of students who earn the corresponding combination of marks. For example, the largest dot in the first graph (near the top right corner) indicate a high number (16) of students who earned 9 out of 10 on question 6 of exam A and 10 out of 10 on question 5.
These graphs show a definite but not very strong correlation between marks earned on Stealer/Mangler questions and marks earned on common type questions. Statistical tests confirm this. For each of the above comparisons we computed:

- the Kendal tau-b (τ b) rank correlation coefficient – tau-b was used because our data contain ties (multiple students with equal marks),
- a p-value indicating the probability that (uniformly) random assignment of the same marks would result in at least as high a correlation.

Here are the Kendal tau coefficients.

Q5A versus Q6A: 0.426	Q5B versus Q6B: 0.4041
Q7A versus Q6A: 0.5139	Q7B versus Q6B: 0.4684

The p-values for all 4 cases are less than 10^{-10}. In other words, they are minuscule.
The p-values confirm definite correlations between how well Stealer and Mangler questions test student ability to code when compared to common type questions. The tau coefficients confirm moderate but not high rank correlations.
We argue that high correlations, to the level where one question can be used in place of another, are not to be expected here. The 3 questions – clubs (Q5), cards (Q6A and Q7B), grades (Q6B and Q7A) – are not similar in that they do not test the same skills. In the same way questions 1-3 (all tracing questions) are also not similar. Indeed the same computation comparing questions 1, 2 and 3 yielded tau coefficients in the 0.3 to 0.4 range.

Conclusions

Posing a programming question as a Code Mangler or a Code Stealer question tests student ability as well as posing it the common way. Furthermore Mangler and Stealer questions can take much less time to grade. Having gathered only one set timing data, our results do not allow us to confidently quantify how much grading time can be saved.
With Code Mangler, the time saved was substantial (over 40%) with the grades question, but much less (about 10%) with the cards question. We think this difference is mainly attributed to the difficulty of the cards question, combined with us asking the TAs to grade that question for correctness even if students chose to write their own code rather than use the mangled lines. Further study is required here.
Qualitative feedback from our TAs tells us that time savings with the Code Stealer is at least as much with the Code Mangler. We were so confident of this that we decided not to collect timing data for the Code Stealer. Some day someone may want to do this.
As with many things in life, posing a good test question is a task that improves with experience and practice. We are firmly convinced that use of Code Mangler and Code Stealer is beneficial. It is a matter of working with them to maximize time savings. So our investigation continues.