Streamlining Grading Toward Better Feedback

Tommy MacWilliam, David J. Malan
tmacwilliam@cs.harvard.edu, malan@harvard.edu

This is CS50.

- lectures
- sections
- walkthroughs
- shorts

This is CS50.

- 8 problem sets
- 2 quizzes
- 1 final project

This is CS50.

- 110 staff members
- 70 teaching fellows (grade problem sets)
- 40 course assistants (do not grade)

Weekly TF Responsibilities

- lead 90-minute section of 12-15 students
- hold 3 hours of office hours
- attend 90-minute staff meeting
- grade 12-15 problem sets

Staff Grading Workflow

- ssh to shared server
- copy, download submissions
- generate PDFs from source code
- annotate PDFs using Acrobat
- record grades in separate spreadsheet
- email annotated PDFs to students

Problems

- too many bottlenecks
- less time spent leaving feedback
- inability to track staff's comments
- unclear if all students read comments

Time Spent Grading

- grading per week: **7.2 hours**
- total work per week: **17.2 hours**
- **42%** of time is spent grading

Goals

- reduce setup time for staff
- audit quality of staff feedback
- understand students' usage

This is CS50 Submit.

- command-line and web uploads
- in-browser source code commenting
- summary comments for submissions
- centralizes submission and feedback

Jane Doe
Here's my submission for problem set 4! I had some trouble with my linked lists, but I think I got everything working in the end.

Problem Set 4

- sll.c
- stack.c
- queue.c

queue.c

```
11  #include <stdbool.h>
12  #include <stdio.h>
13  #include <stdlib.h>
14  #include <string.h>
15
16  // the capacity of the queue
17  #define CAPACITY 10
18
19  // a queue
20  typedef struct
21  {
22      // the index of the first element in the queue
23      int head;
24
25      // storage for the elements in the queue
26      char* strings[CAPACITY];
27
28      // the size of the queue
29      int size;
30  }
31  queue;
32
33  // declare a queue (as a global variable)
34  queue q;
35
36  /**
37   * Puts a new element into the queue into the "end" of the data structure
38   * so that it will be retrived after the other elements already in the
39   * queue.
```

John Smith
Great use of a constant here!

John Smith
Hmm... do we really need to use a global variable here? Why might this not be the best design?

Results

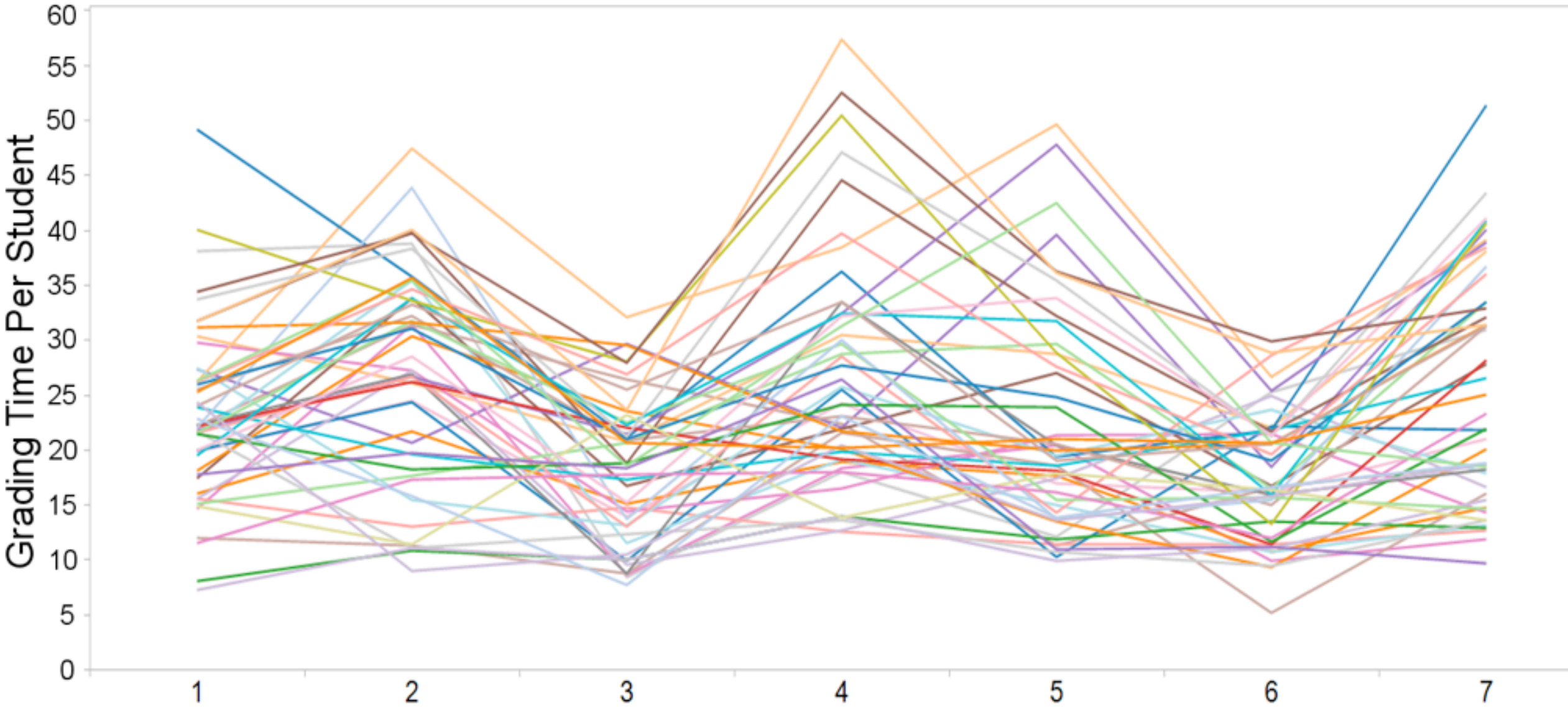
- staff efficiency
- feedback quantity
- students' usage

Staff Efficiency

- 10% fewer hours grading per week
- 13% fewer minutes grading per student
- independent of prior teaching experience

Staff Efficiency

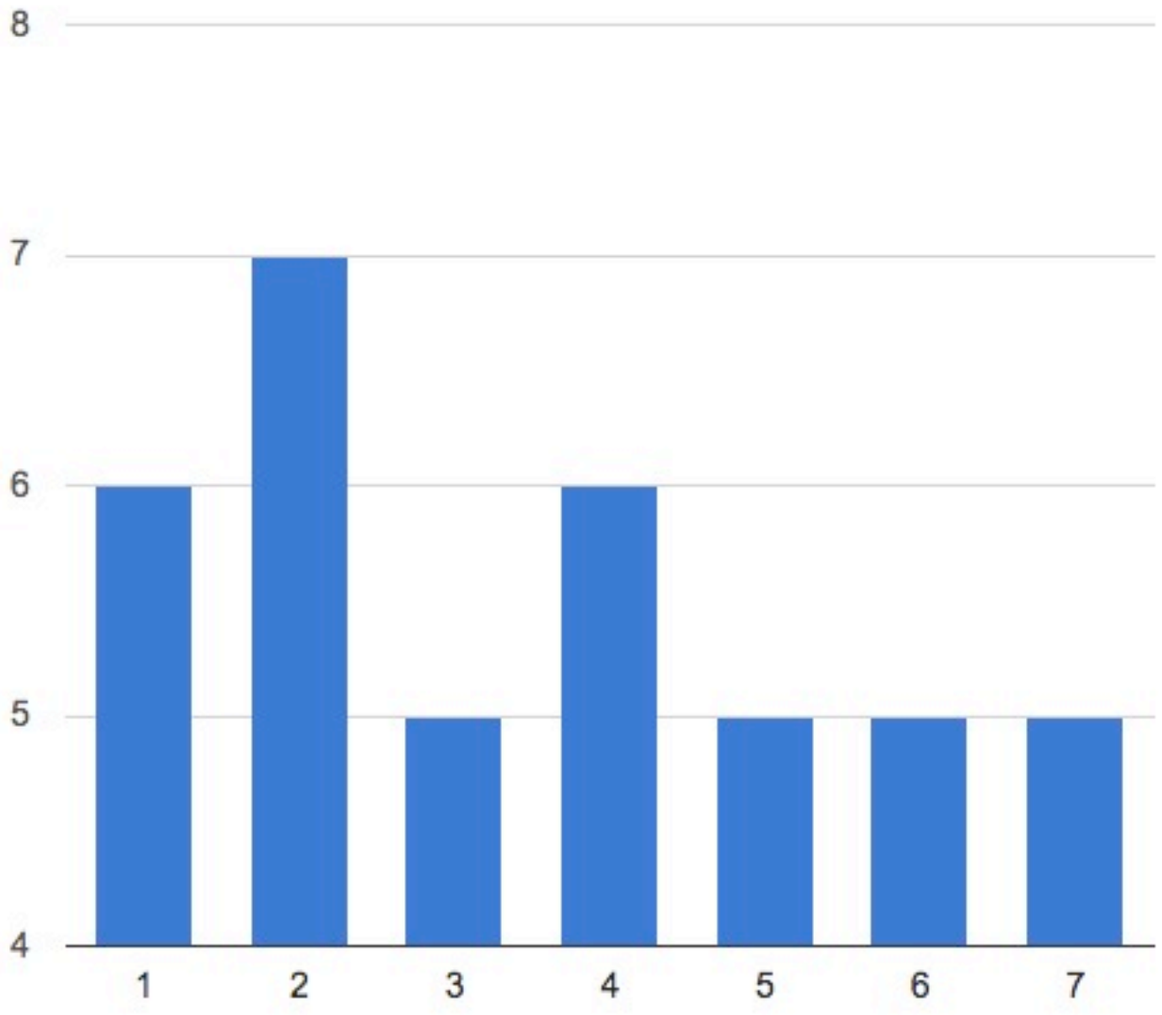
- fewer reported grading bottlenecks
- 136% more time spent on discussion board
- locate problem set-specific bottlenecks


Grading Time Per Student
60
55
50
45
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7

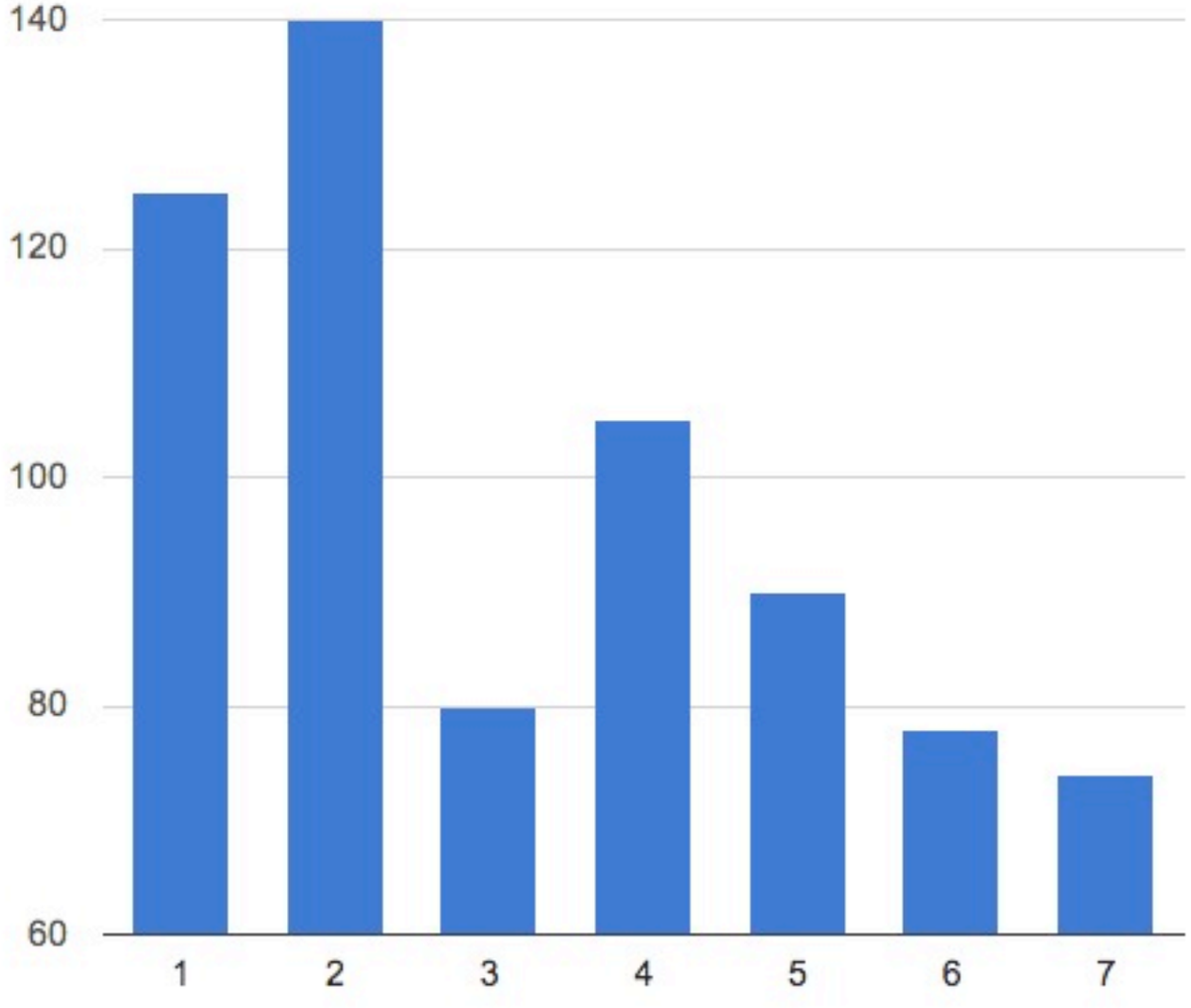
Feedback Quantity

- median number of comments per student remains constant
- median number of words per comment declines

Median Number of Comments

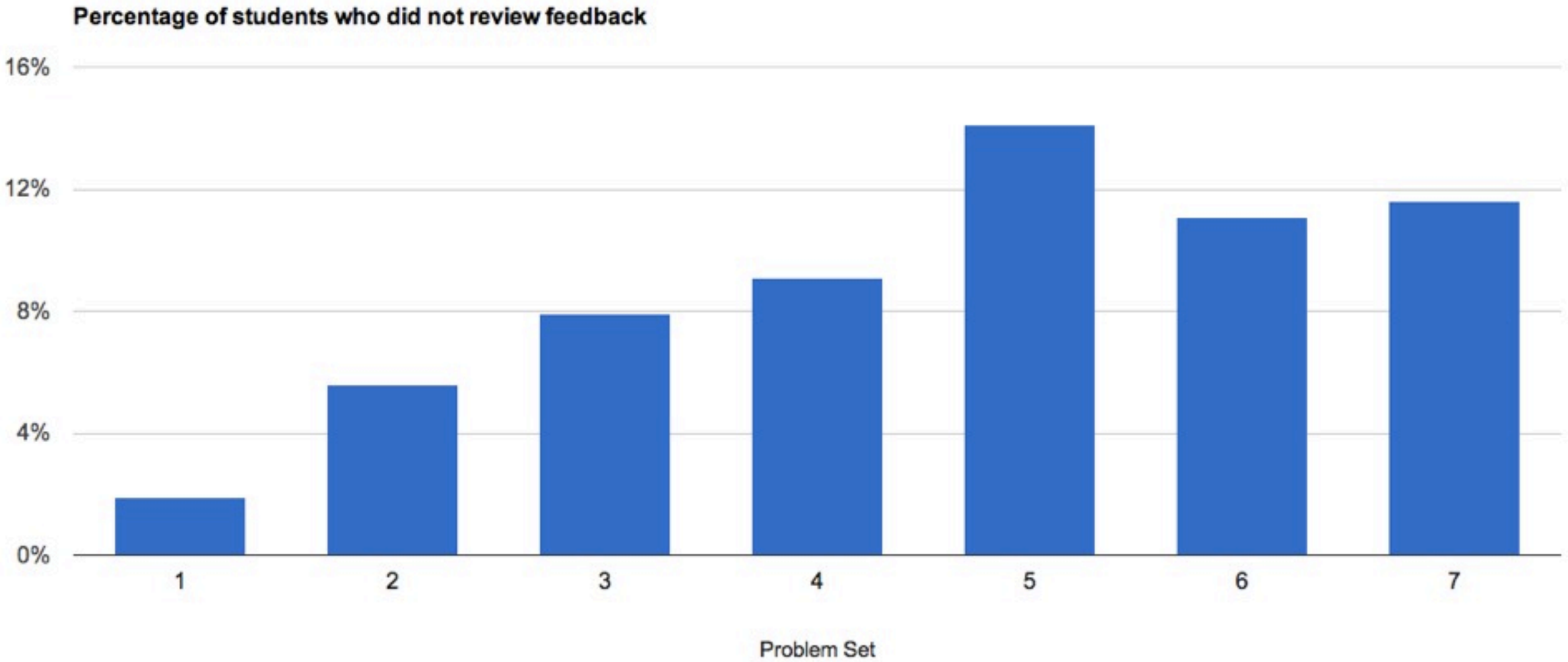


Median Comment Word Count



Students' Usage

- on average, **9%** of students never open feedback
- mid-semester peak of **14%**
- number of students who do not open feedback increases over time


Percentage of students who did not review feedback
16%
12%
8%
4%
0%
1
2
3
4
5
6
7
Problem Set

Questions