15295 Fall 2018 #6 Problem Discussion
October 3, 2018

This is where we collectively describe algorithms for these problems. To see the problem statements follow this link. To see the scoreboard, go to this page and select this contest.

The theme of this contest (except for the last problem) was using bit operations.

A. Palindrome Pairs

We convert each string into a *signature*. A signature is just a 26-bit string representing 26 lower case letters. For each character in the string, we flip the corresponding bit once. In other words, letters that occur odd times will have a 1 bit, and that occur even times will have a 0 bit.
We then maintain a hashmap M: signature -> count (int). It is easy to see that if two strings have the same signature, they form a pair. But, if two signatures only differ by a single bit, they also form a pair -- you can have a single occurrence of a letter in the middle of the palindrome.
Finally, we loop through all input strings. For each string, calculate its signature *S*, find *S*'s count in M, accumulate it to the answer. Then also find the 26 possible signatures that are only 1-bit different, and accumulate the count to the answer. Lastly increment the count of *S* itself. Complexity is O(n).
-- Edmond

B. Bicolorings

C. Party Lemonade

D. Careful Maneuvering

E. Random Task

F. The Shortest Statement