University of Waterloo
CS240 Fall 2017
Assignment 2 Post Mortem

Problem 1
Many students did not include the base case in part a

Some students simply stated, rather than showed, that the mean is equivalent to the \(\frac{(n-1)}{2}\)th element in part b

In part c many students used the example of all the elements being the same, which relies on the assumption that the algorithm uses a inefficient way of dealing with duplicate elements. For full marks students should have given an example which did not rely on that assumption.

Problem 2
Some students provided examples that were not \(O(n \log \log n)\). Others provided examples that achieved the desired time complexity, but did not completely sort the array (e.g. assuming that each element in a block of size \(\log n\) is greater than every element in the previous block and just sorting each block separately).

Problem 3
Parts a and b were generally well done.

For part c, some students provided a recurrence relation, but did not solve it. Some students tried to express \(T(n)\) as a \(\sum_{k=1}^{\infty} f(n, k)\) where \(k\) is the depth of recursion, but did not give the recurrence relation.

Problem 4
Part a was poorly done. Many students gave the answer of \(n - 1\). Often, students proposed a specific algorithm and calculated a lower bound for it, or assumed that all coins have to be compared individually to another coin at least once, but did not prove why.

Parts b and c were well done, though some students did not include a runtime analysis with their algorithms.
Problem 5

Part a was well done, though some students just mentioned the need to bubble down, while if the element swapped in is larger than its new parent, we would need to bubble-up instead.

Many students did not submit code for part b. Some students submitted code that did not compile.
A common mistake in the implementation of removeMaxWins and removeMinLosses was to only remove the team from the max or min heap, respectively, instead of removing the team from both.