

Homework 0 Writeup

Alvin Lin

March 21, 2018

Problem 0

For this problem, I used the Sieve of Eratosthenes to filter out all non-primes from 1 to n , using an $(n+1)$ -sized array and the array index to represent the number. This algorithm starts with two, filters out all multiples of 2, moves to the next number that has not been filtered out, removes all of its multiples, and so on. The numbers that are left at the end of this process are guaranteed to have no multiples, hence they are prime. This algorithm has a time complexity of $O(n^2)$.

Problem 1

For this problem, I maintain two integers representing the lowest and second lowest number. Then, we iterate through the given numbers. If the current number is lower than the lowest stored number, then we replace it and the lowest number becomes the second lowest. Otherwise, if the current number is lower than the second lowest stored number, then we simply replace it. This has a time complexity of $O(n)$ since it only needs one pass to determine the lowest and second lowest numbers.