

CompSci 251: Intermediate Computer Programming

Lab 8 – 2018

You are going to write a program which implements the Sieve of Eratosthenes. The sieve removes all numbers from a list, from 2, 3, 4, . . . , n, that are not prime numbers. Below is a link to the Wikipedia page which shows an animation of how the program works.

https://en.wikipedia.org/wiki/Sieve_of_Eratosthenes

Below is a description of one way to do this. (If you can think of another way to perform this task, you may do so. But, the end result must still match the output below and you must use an ArrayList!)

1. Create a list of integers from 2 to n (2, 3, 4, . . . n).
2. For integers p in the range [2, n/2], do the following:
 - a. If p is crossed off, skip the following step
 - b. Otherwise, cross off all multiples of p in the list not including p itself. Some of these numbers may already be crossed off and that is ok. (ie. 2p, 3p, 4p, . . .)
3. Each number in the list that wasn't crossed off is prime.

For example, if we run this algorithm on n = 10, we get the following steps:

2 3 4 5 6 7 8 9 10	initial list
2 3 4 5 6 7 8 9 10	cross off multiples of two
2 3 4 5 6 7 8 9 10	cross off multiples of three (six already crossed off, nine now crossed off)
2 3 4 5 6 7 8 9 10	skip, four is already crossed off
2 3 5 7	result – all are prime numbers

You should implement this method using an ArrayList of Integers. You can simulate crossing out a number from the list by removing that element from the list.

Be aware that ArrayList implements two different remove methods. remove(int) will remove an element at the index specified by the argument, and remove(Integer) will remove the first element equal to the object specified by the argument.

You are responsible for writing a driver that will match the following output. If you are off by a phase that is ok.

```
Please enter a number n: 15
Initial: [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]
Phase #1: [2, 3, 5, 7, 9, 11, 13, 15]
Phase #2: [2, 3, 5, 7, 11, 13]
Phase #3: [2, 3, 5, 7, 11, 13]
Phase #4: [2, 3, 5, 7, 11, 13]
Final:   [2, 3, 5, 7, 11, 13]
```