

A3-A Soft Landing

Stuart Thiel

July 31, 2022

Introduction

This assignment is worth **5%** of your grade. This is an **individual assignment** (so individual, I put *your* student ID on it) and you should not share your assignment with anyone else. The **assignment is due August 7th at 11:59pm, Montreal time.**

All submissions must go through EAS: <https://fis.encs.concordia.ca/eas/>

After you submit, go back to check EAS to confirm that you just uploaded your assignment!

Q1) A Prime Example

Write a Class called Primes. It should have two fields: `int[] primes` (starting with 2,0,0) and `int size` (starting at 1). It should also have a method `printPrimes(int n)` that prints the first n primes, one on each line.

`printPrimes` should work by printing its known primes, if it has enough of them.

If it has to print more than are currently known in the primes array (for example, initially it only knows about "2"), then the method must search each integer larger than the last known prime until it finds another prime. It should add this new prime to the end of the array. When the array is full, a new array with twice the size should be created and replace the old array, with all the old values copied over and zeroes in other positions. To find a new prime, check that the current number is not divisible by each of the known primes that you have (use a for loop over the know primes you have). You may (and probably should) create a few new methods do do various tasks.

Write a main method that calls `printPrimes` with 2 and then 10 to validate that it is working.

Q2) Making Censor of File IO

Write a Class called Censor. It should have a static method `censor(File source, File destination, File badlist)` that reads each line in `badlist` and then replaces any occurrence of a word (or words) specified on a line in `badlist` in the `source` file, putting the new text in the `destination` file. The replacement should consist of the same number of `*` characters as the replaced text length. It should not match across linebreaks.

You can (and should) use anything in the String class, anything related to regular expressions and your preferred File IO from the `java.io` package. You should code everything else on your own.

Your submission should include an appropriate text chosen by you, no longer than 20k, an amusing badlist file and the resulting output in a third file. You should have a main method that uses the first two files to generate the third.

Example: with badlist "foo", "bar" and input "The prof uses foo and bar too often in class. It is like being in a barn." would be output "The prof uses *** and *** too often in class. It is like being in a ***n."