Assignment 0 - Github Familiarity
Due: Friday, January 14, 11:59pm

Program

You are to implement a program called *filesize*. The *filesize* program takes any number of command line arguments. It first prints the name of the program. Then, for each argument, it should open the file, count the number of lines in the file, and print the number of lines that it finds. If the file cannot be opened, it should print -1 for the number of lines.

Example

For example, when run as follows, your program should produce output in exactly this format:

```
BSH:Saru> ./filesize input.3lines
program: ./filesize
 input.3lines: 3

BSH:Saru> ./filesize
program: ./filesize

BSH:Saru> ./filesize input.notafile input.unreadable
program: ./filesize
 input.notafile: -1
 input.unreadable: -1

BSH:Saru> ./filesize input.3lines input.1line input.notafile input.unreadable
      input.2linesNonewline  input.annoying\ file\ name  input.just8blanklines
program: ./filesize
 input.3lines: 3
 input.1line: 1
 input.notafile: -1
 input.unreadable: -1
 input.2linesNonewline: 2
 input.annoying file name: 3
 input.just8blanklines: 8
```

Requirements

The program must be written in c++. The github repository contains a headstart filesize.cc file, a Makefile (which you should NOT change), and the testing input files. Your solution must use only the file filesize.cc

Your output must be identical to the output shown in this document, including the spaces.

You must NOT change the Makefile. Programs that do not compile with that Makefile will receive zero credit.

This is a trivial program to write. If you file solution file is more than 30 lines longer, you're probably making it too complicated!

Part of your grade fill be on good coding style in filesize.cc, but this exercise is primarily intended to make sure that you can correctly use the github environment to submit and test subsequent programs.

Turn In

The program will be turned in using Github for Education. We'll talk more about that in class.