CS 2204 Lab 6

your name here (please print):

your student ID number here:

WARM UP:

1. Create a subdirectory called `lab6` under your home directory.
2. Inside this directory, create several files, each with some random text in them. Make sure some files have extension (i.e., names ending with) `.c` and some files have extension `.test`.

QUESTIONS TO ANSWER:

In this lab assignment, we are going to explore the programming capabilities of `gawk`. Instead of writing `gawk` scripts to process a file's contents, we are going to write `gawk` scripts to process the output of the `ls -l` command. For instance, we will invoke `gawk` as (make sure you are `cd`-ed into the `lab6` directory):

```
ls -l *.c | gawk -f <your gawk script here>
```

1. (3 points) Write a `gawk` script to print the sizes of files, one on each line. First, analyze a typical output of the `ls -l` command to see which field contains the size information. Then, determine what the field separator is and whether the default field separator of `gawk` will work, or if you have to specify it using the `FS=` command. Write down your `gawk` script here:

2. (2 points) There is a potential problem with the script you wrote above. If we had invoked the `gawk` command as:

   ```
   ls -l | gawk -f <your gawk script here>
   ```

   note that we have an extra line in the `ls` output, beginning in `total`, that will mess things up. You must tell `gawk` to ignore this line. The way you do it is using a test on the variable `NF`. Do a `man gawk` to see what `NF` means. Then, write a `gawk` script that uses a pattern to check if the `NF` variable is 9 (using the `==` operator) and only if true, proceed to print the size information (Why is 9 the magic number?). Write your new script below.

3. (2 points) Oh, Geez! There are even more problems. What if the output of `ls -l` involved a directory or a symbolic link? You may not have such issues in your `lab6` subdirectory, but we should try to write as fool-proof a command as possible. In this case, we want to ignore lines corresponding to directories or links. So, in addition to the `NF` test, check to see if the line involves a file. Use a logical AND (&&) to achieve this effect. Only after both conditions are satisfied, proceed to print the size information. Write your new script below.

4. (3 points) Let us suppose we actually want to find the average of the sizes of the files, and just print that. Rewrite your `gawk` script to achieve this effect. To convince yourself that the script is general enough, use it to first find the average size of files ending in `.c` and then, of files ending in `.test`. Write your script below.