HW 3: Bash Scripting

Due: October 31

Overview: The last homework had you create a pipeline of utilities to locate files with the same content. This homework will have you perform this task again using features of `bash`.

Objective: Write a bash program that finds all the files with the same md5sum hash, sorts them by the size of the group sharing that hash, and prints out the groups as a comma-separated list.

Guidelines:

- You *must* do this by splitting the program into three functions:
  - The first function, `find_and_count_duplicates` , will be for listing the files (find), determining their md5sum (md5sum), and creating a few arrays. One array will map the hash to all the files that have that md5sum. Another array will store a count of the number of duplicates for each md5sum.
  - The second function, `assign_md5sums_to_group_sizes` , will have you create an array that will store all the md5sum hashes for a single group size.
  - The third function, `print_group_sizes` , will be to iterate over the group sizes, and print the files for each md5sum group of that size.
- You must comment some of your code in each function (use `#` before text).
- You must use at least one `for` loop and at least one `while` loop.

Restrictions:

- No credit will be given if you use the method from solution 2.
- The only commands you are permitted to use outside of bash's builtin functionality are `find`, `md5sum` , `sort`, and `tr` (once each).

Helpful hints:

- Command substitution is one way of initializing an array
- To use a parameter as the key of an associative array, you will need to quote that variable.
- Printing after each step to diagnose problems early on. Colorizing the input can also help you debug.

Bonus 1: Use the ANSI escape codes to print out in color. This will make output much easier to read. Hint: These look like

```
# Regular Colors
Black='\033[0;30m'        # Black
Red='\033[0;31m'          # Red
Green='\033[0;32m'        # Green
Yellow='\033[0;33m'       # Yellow
Blue='\033[0;34m'         # Blue
Purple='\033[0;35m'       # Purple
Cyan='\033[0;36m'         # Cyan
White='\033[0;37m'        # White
```

Bonus 2: Print out the directory of the files in one color, and the actual filenames in another color.

Bonus 3: Is this faster or slower than the first homework 2 solution? Compare the time taken for both on different numbers of *unique* files using the `time` command. Test for 5000 unique files, 20000 unique files, and 50000 unique files. You may use the command:

```
for i in `seq <filecount>`; do echo $i >> $i; done
```

to create unique files.

Submission:
Upload a single text file (.txt) containing all of your answers to Canvas by the due date.