CS360 – Operating Systems
Assignment 1

Objective

This is a warm up assignment to help you practice writing and building programs in C. It will also introduce you to 2 commonly used UNIX/LINUX utilities that Linux users use on a daily basis (you probably already used them or heard of them).

In this assignment you will create simplified versions of two commonly used UNIX/LINUX commands: `cat` and `grep`. Both cat and grep can take many options. You don't have to support all options. You will just support the most common and basic functionalities of these 2 utilities.

Instructions

This project was borrowed from Prof. Remzi Arpaci-Dusseau Operating Systems class at the *University of Wisconsin – Madison*.

Operating Systems: Three Easy Pieces
Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau
Arpaci-Dusseau Books
August, 2018 (Version 1.00)

Instructions (and implementation hints) for completing this assignment are found here. The instructions call for writing 4 utilities: cat, grep, zip, and unzip. **You do not have to do zip and unzip. You only need to do cat and grep**.

Instead of calling the utilities `wcat` and `wgrep` (w is for Wisconsin), let's call ours `bccat` and `bcgrep` (bc for Bellevue College).

Before writing code, learn and experiment with how the "real" `cat` and `grep` work. I included a sample file ("test.txt") to experiment with. Put test.txt in some folder, cd to that folder, and practice with cat and grep:

```
prompt> cat test.txt
```

and

```
prompt> grep some_word test.txt
```

You will see that `cat` prints the content of test.txt to standard output (the terminal window). And `grep` searches test.txt for lines that contain some_word and prints those lines to standard output (lines that don't have some_word are ignored).

What You Need to Turn In

Your folder structure should look like the below (blue are folders, green are files):

```
john_smith_hw1
        bccat
                bccat.c
                makefile

        bcgrep
                bcgrep.c
                makefile
```

Each subfolder (bccat and bcgrep) contains a *makefile* for its corresponding program. The executable programs should be called **`bccat`** and **`bcgrep`**. For example: `gcc bccat.c -o bccat -Wall` can be used in the makefile to build `bccat.c` into `bccat`. If you are not familiar with how to author a *makefile*, look in the C Tutorial included with Module 1.

I will build your programs by running `make`. Verify that `make` works and your programs build without errors. If I get compilation errors, you will lose points and I will return the program to you to fix.

Test your `bccat` and `bcgrep` thoroughly before you submit.

Zip john_smith_hw1 to generate a zip file: john_smith_hw1.zip
Upload john_smith_hw1.zip to Canvas (do not send it to me by email).

Late Submission Policy

- Up to 2 days late: 30% penalty.
- More than 2 days late: Not accepted. You will lose all points on the project.