**CS2750 System programming and Tools Fall 2019**

**Project #7 [40 points]**

---------------------------------------------------------------------------------------------------------

Due date is Tuesday, Dec 10th. This file should just be one executable, with a makefile to compile it.

**Task:**

In this project you will be writing a program that will copy itself with fork, then do a cat of a file given as a command line argument. Your original copy should use a “wait” call to wait until the cat command has completed, at which point it should output a message and terminate. I want you to have text output at each significant step. Your executable should be called “catwrapper” and would be called as follows, assuming we had a file called dog.txt that contained the word “Star”:

./catwrapper dog.txt

Then the expected output would be something like:

In process pid 3126 (parent) before executing cat command.

In process pid 3128 (child)

Now process pid 3128 (child) executing cat command

Star

In process pid 3126 (parent) after wait system call, now terminating

Note that you should test the error status from fork and also even exec (though if in the child code you get past the exec, that always indicates a problem).

Do NOT use loops in your program at all. There is no reason for a loop in this code and using loops your first time using fork can lead to bad things.

**Details:**

You MUST use a call to fork(), then the child does an exec() family call while the parent does a wait().

You only need one source file, as discussed. However, you also must have a makefile that will compile the project for me. If you do not include a makefile your project will not be run.

I want you to submit an example of the output of your code, as a screenshot.

**Submission:**

Submit the source file, your makefile, and the screenshot on canvas