ETEC3702 – Concurrency
Lab 8 – Inter-Process Communication and Pipes
Due date: 7 April 2020 by the end of the day.

For this lab you are going to write a program that uses multiple processes and pipes to communicate between them.

Most operating systems allow processes to be launched and linked with pipes from the command line.

A command line illustrating this in UNIX would be:

$ cat file.txt | grep "Yost" | wc -l

Which will launch 3 processes:

P1: cat – cat copies the contents of the specified file to stdout.
P2: grep – grep filters the input stream (stdin) for the specified pattern "Yost".
P3: wc – word count counts the number of lines.

The command interpreter starts the programs in separate processes, creates the pipes, and connects each pipe-end to the processes' stdin or stdout as specified.

Assignment:

We're going to write a Python program that simulates this by creating the following functions:

```
def cat(filename,pipeout):
    #function opens the file specified and loops by reads a line at a time,
    # then writing each to the pipe specified by pipeout.
    # once all lines have been read, the function should write None
    # to pipeout then terminate.

def grep(pattern,pipein,pipeout):
    #Loop
    # read a line from pipein
    # if the line read is None then write None to pipeout and terminate
    # if the pattern specified in contained in the line
    # then the line is written to pipeout.
    # lines not containing the pattern are discarded.

def wc(pipein,pipeout):
    #initialize a count variable to zero
    #Loop
    #read a line from pipein
    # if the line read is None then exit the loop
    # otherwise increment the count
    #after the loop, write the count to pipeout as a string: "Lines:"+str(count)
    #write None to pipeout and terminate

def printer(pipein):
    #Loop
    # read a line
    # if the line is None then terminate
    # otherwise print the line out using print.
```

Write a main program that creates pipes and launches all of the process with the pipes interconnecting them as follows:

Program 1: `cat "sonnets.txt" | grep "thee" | printer`
Program 2: `cat "sonnets.txt" | grep "thee" | wc | printer`

Note that in your main program:

- You'll have to create the pipes to be passed to the functions.
- You'll need to start all of the processes and wait for them to complete.
- You may want to an input("Enter to Exit") at the end of main.
- You should remember that Idle won't show output from sub-processes so you should run the programs directly.