Denzel Mathew
COP4600
Section# 1089
10/29/2015
Term Project: Part1

The work submitted is my mine and the Honor Code was neither bent nor broken.

Summary: The educational objectives of this assignment were to get used to demonstrate inter process communication through the use of piping between processes, reading man pages and working in the openBSD environment. This part helps set the foundation for all the future parts of this term project as it allows us to navigate around openBSD as well as utilizing the man pages in terminal (or online) as reference.The hardest part for me was trying to figure out why I was getting multiple forks under receiver or sender (depending on the order at first). I eventually figured out why by drawing out the process tree with parent and going branch by branch where the fork is happening. This issue was what was causing my pipe to not write or read properly. Once I got this fork issue fixed, my pipe was working accordingly. I was debugging this by going through various print statements to identify what my process ID was for each process under the different sender/receiver sections. What helped me the most about this project were the man pages. I now have a greater appreciation for these manuals as they explain what the functions do and what parameters they require. Pretty much everything about the function is explained in the man pages. Easiest part was probably setting up VMware for openBSD and testing my programs within it. I decided not to use emacs to manually enter my code as it was slow and bothersome, so I scp'ed from my terminal to the virtual machine. It was so quick and painless and I recommend it to anyone. What could be done in the future as I noticed with my friends frustration (though not sure why) is an instruction set for using file transfer protocols (FTP).

1. The program compiles without errors.
2. The program compiles without warnings.
3. The program runs without crashing.
4. I tested this program by running it in the openBSD environment and observing the output generated.
5. This program meets all expectations.
6. There are no known bugs.
7. The program runs correctly.