CSCD 340
Lab 4

1. Modify cscd340lab3prob3.c so every command entered on the command line is entered into your linked list from homework 1. Your data must be void * in your linked list. If it is not a void * then you will receive 0 points for this problem.
   - You will need to create a data structure named commands. This data structure will hold an int for the command number and a char * for the actual command.

     ```
     struct com
     {
        int num;
        char * theCommand;
     };

     typedef struct com commands;
     ```

   - Modify the while loop so that if the user enters print it prints out the items in the list in the following format:

     1 ls -l
     2 pwd
     3 asdfgh

   - Make sure that at the end of the program you clean up all the memory of the linked list.

   Run your program and save the output into a PDF named cscd340lab4, verify that your linked list is properly working. Rerun and use valgrind to verify you are leak free. Also save the valgrind output in cscd340lab4 – ensure you identify the problem.

2. In class we discussed a program to duplicate the functionality of the command line input
   ls –al | wc –w

   as an illustration of the PIPE, FORK, and DUP system calls. Design, implement and test this program. Name your C file cscd340lab4prob2.c

   a) Execute the command at the command line first and record the output.

   b) Run your program and verify your output is the same as the recorded output.

   Put your output in the pdf file named cscd340lab4 – ensure you identify the problem.

3. In cscd340lab4prob3.c main, there is nothing fancy here, meaning I should be able to enter ls –l | wc –w and have it display the output from ls –l being piped into wc –w.

   NOTE: This is a onetime execution and you will need some way to figure out what is on the left side of the pipe and what is on the right side of the pipe

   Include in PDF, at least 3 output runs of your code with 2 different pipe commands. Include a valgrind run of your program, and answer this question since exec changes the process image does your program leak memory? Justify your answer.

4. Copy cscd340lab4prob3.c and named it cscd340lab4prob4.c. You will need to modify pipeIt so the executable does not exit after the pipe code executes. HINT: double fork in pipeIt. There is nothing fancy here, meaning I should be able to enter ls –l | wc –w and have it work and then return to main and allow the user to enter another command.

   Include the PDF, that contains at least 3 output runs of your code with 2 different pipe commands, and one no pipe. This will be a single run of the program. Include a valgrind run of your program.

TO TURN IN

A zip

- Your C files
- Your PDF files
- Make file targets
  - all
  - prob1
  - prob2
  - prob3
  - prob4

Your zip is named your last name first letter of your first name lab4 (Example steinerslab4.zip)