Faculty of Engineering and Applied Science
Department of Electrical, Computer and Software Engineering

SOFE 3200U - Systems Programming
Lab 4

Task 1: Random Phrase

Using system calls, create a random phrase and print it to standard output. You may use standard library functions for random number generation.

Your phrase may be a sentence in the following format:

> The <adjective> <adjective> <noun> jumped over the <adjective> <noun>.

If you wish, you may change the sentence and add more word types, but you must have at least two word types (ex: adjective, noun).

Task 2: Producer-Consumer

Background

In this task you need to implement a common design pattern called producer-consumer. This occurs when you have one process generating (producing) data, or tasks to perform, and another process reading (consuming) the data, or performing tasks.

The program you create will read digits from an input file and add to a total in a fault-tolerant manner. This is accomplished using a producer that produces sanitized data.

Deliverables

1. Create a program that will take two arguments: 1) an input file; 2) an output file.
2. The program must create a child process to read the input file line-by-line. Each time the child reads a line, it will remove all non-numeric characters from the line.
3. The new line must be sent to the parent process through a stream created using the pipe() system call.
4. The parent process will read all the numbers from the child, convert them to an integer (you can use atoi for this), and compute a sum.
5. Additionally, at the beginning of your program, print each file descriptor (there will be 2) created after calling pipe(). You may use printf for this.

Task 3: Program Tracing

Background

- The system utility `strace` uses this system call. You need only provide one argument to strace, which is the program to trace.

Deliverable

1. Run two applications (Application 1 and Application 2) each for one minute. Identify their respective PID to track them. Write a shell script which uses strace to count the system call “open()” and “close()” for both Application 1 and Application 2.

Lab 4 Deliverables

Submit the code and the report. This lab consists of 10 points. The marking details are:

1. 70% for the report and code

The report should contain following things:

· A title page that includes the title of the lab, date and group members
· Objectives of the Lab
· Verification, steps, and guideline of the code implementation
· Individual roles, responsibilities, and contributions
· Submission of code

2. 30% for answering the questions

In every lab, one random person from each group will be asked random questions from the previous lab and the person changes in every week. Based on that students answer the whole group's marks will be given.