Lab 10

1. Write a C program that receives a filename as a command line argument and creates a child process. The file will contain only integers, the first number, N (0 < N < 100, so the entire content of the file can be loaded in memory) will represent the total number of numbers in the file, excluding itself. The child process will read the numbers from the file and send them to the parent. The parent will display N first and will await input from the user. The user will input an integer number X and the parent process will print the first X numbers (or N, if X > N). If numbers are left, the parent process will prompt the user to input a new value for X and the parent will print the next X numbers. The process ends when there are no numbers left to print or when the user inputs a number less or equal to 0.

2. Write a C program that receives N (0 < N < 10) command line arguments. The main process creates N child processes. The K-th child process will count all the letters and all the digits of the K-th command line argument and will send the two counts to the parent. The parent will calculate the sum of the letter and digit counts it receives from the children and print the two totals.

3. Write two C programs (A, B). Process A reads a string (maximum 100 characters) from standard input, removes the last X characters from the string, where X is randomly generated (0 < X < 4), and sends the new string to process B via FIFO. Process B also removes the last X characters from the string (X is generated with the same rules as above) and sends the new string back to A. The algorithm repeats until one process removes the last character from the string, at which point it prints a message and both processes terminate.

4. Write two C programs that communicate via FIFO. One program (A) will generate a random number N between 0 and 1000. The other program (B) will try to guess the generated number. B will start by generating a random number between 0 and 1000 and send it to A which will respond with -1 if the “guess” is lower than N, 1 if the “guess” is greater than N, and 0 if the guess is correct. B will continuously restrict the guesses based on the responses from A and will stop once it correctly guesses the number. Program A will create and destroy the FIFO.