CS344: Operating Systems Lab

Lab # 04 (1 Questions, 90 Points)
Held on 12-Sep-2023
Lab Timings: 09:00 to 12:00 Hours Pages: 3
Submission: 12:00 Hrs, 12-Sep-2023
Instructor Dr. V. Vijaya saradhi
Head TAs Adithya Moorthy & Laxita Agrawal
Department of CSE, IIT Guwahati

a. This assignment is based on chapter 3, `Process Management` in the book `Operating System Principles, Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne`.

b. In order to perform this assignment, understanding of system calls `pipe(), dup2(), fork(), exec(), write()` are essential.

c. Carefully read the manual pages for the usage of above system calls.

Question 1: (90 points)

Interprocess communication: Implement the following:

a. **pipe**s

i. (5 marks) Write a 1^{st} C program to compute the following sequence.

$$n = \begin{cases} \frac{n}{2} & \texttt{if n is even} \\ \\ 3 \times n + 1 & \texttt{if n is odd} \end{cases} \quad (1)$$

The input n should take a default value 100. Name the executable of this program as `compute_sequence_1`

ii. (5 marks) Write a 2^{nd} C program that reads one integer at a time from standard input till value 1 is reached. Computes the sequence given in equation (1) for each number this program read. Name the executable of this program as `compute_sequence_2`

iii. (30 marks) Write a 3^{rd} C program with the following specifications:

- Your program waits for user to input a string which is a command of the form `./compute_sequence_1 | ./compute_sequence_2` If the user enters "quit" your program should terminate.
- (10 marks) Parse the input to identify the given two `commands`.
  `left hand side of pipe: ./compute_sequence_1;`
  `right hand side of pipe: ./compute_sequence_2;`
- (20 marks) Your task is to write the output of `./compute_sequence_1` to pipe and execute `./compute_sequence_2` by reading the input from the pipe. To achieve this, perform the following:
  i. (2 marks) Create a pipe

ii. In the main, perform the following:
   A. (2 marks) Close the read end of the pipe
   B. (2 marks) Copy the write end of the pipe to the standard output file descriptor `STDOUT_FILENO`
   C. (2 marks) Close the write end of the pipe
   D. (2 marks) perform the program execution `compute_sequence_2`

iii. (2 marks) In the child, perform the following:
   A. (2 marks) Close the write end of the pipe
   B. (2 marks) Copy the read end of the pipe to the standard input file descriptor `STDIN_FILENO`
   C. (2 marks) Close the read end of the pipe
   D. (2 marks) perform the program execution `./compute_sequence_1`

Explanation: left side command of | will compute the sequence for 100 and produces the numbers say {`50, 25, 76, ...`}. Write these numbers into a pipe. Right side command of the | should read each of these numbers one by one from pipe and produce sequence corresponding to number `50` followed by sequence corresponding to number `25`, sequence corresponding to number `76` etc.

iv. (50 marks) Write a 4^{th} C program with the following specifications:

- Your program waits for user to input a string which is a command of the form `command_1 | command_2 | command_3`. If the user enters "quit" your program should terminate.
- (10 marks) Parse the input to identify the given three `commands` and their associated arguments.
- (5 marks) Your task is to write the output of `command_1` to pipe.
- (20 marks) `command_2` reads input from pipe, executes `command_2` and writes output to pipe.
  5 marks for `command_2` reading input from pipe.
  15 marks for `command_2` writing output to (a new) pipe.
- (15 marks) `command_3` reads input from pipe, executes `command_3` and writes output to standard output.

i. Example input: `ls -l | grep pdf | grep Mar`
```
command_1:  ls -l
command_2:  grep pdf
command_3:  grep Mar
```

ii. Example 2: `ls -al | grep -v total | cut -d' ' -f1`
```
command_1:  ls -al
command_2:  grep -v total
command_3:  cut -d' ' -f1
```

iii. Example 3: `ls -l | cut -d'r' -f2 | grep w-`

```
command_1:  ls -l
command_2:  cut -d'r' -f2
command_3:  grep e
```