COMP-8567
Assignment 03
Winter 2024
Due Date: Mar/23/2024
50 Marks

Write a C program **shell24** (**shell24$**) that goes into an infinite loop waiting for user's commands. Once a command is entered, the program should assemble and execute each command using fork(), exec() and other system calls as required with the following rules and conditions.

Note: A modular approach would be helpful.

Rule 1: the program/command **newt** (**shell24$**newt) must create a new copy shell24 //There should **not be any upper limit** on the number of new shell24s terminal sessions that can be opened

Rule 2: The argc (includes the name of the executable/command) of any command/ program should be >=1 and < =5

Examples:

- **shell24$ date (argc =1)**
- **shell24$ ls -1 -l -t ~/chapter5/dir1 (argc =5)**
- **shell24$ cat input1.txt input2.txt (argc=3)**

Rule 3: The argc of induvial commands or programs that are used along with the special characters listed below should be >=1 and <=5

- Ex: **shell24$ ls -l -t |wc** //the first command has argc=3 and the second command has argc=1 which are used along with the special | character

Special Characters

The program should handle the following special characters (In accordance to Rule 3 and the additional rules listed below)

- # Text file (.txt) concatenation (upto 5 operations)
  Ex **shell24$ check.txt # new.txt # new1.txt # sample.txt**
  // Files must be concatenated in the order in which they are listed and the final result is displayed on the stdout
- | **Piping** (up to 6 piping operations should be supported)

  Ex shell24$ ls |grep *.c|wc| wc -w

  // Every command/program can have argc >=1 and <=5 as per Rule 2
- >, <, >> **Redirection**

  Ex: shell24$ cat new.txt >>sample.txt

- && **Conditional Execution** // up to 5 conditional execution operators should

  be supported and could possibly be a combination of && and ||

  Ex : shell24$ ex1 && ex2 && ex3 && ex4

  - **shell24$ c1 && c2 || c3 && c4**
- || **Conditional Execution** // see &&
  - Note in both && and ||, the argc of each command should be >=1 and <=5 as per Rule 2

- & **Background Processing**
  - **shell24$ ex1 &** //should run ex1 in the background
  - //Note: **shell24$fg** should bring the last process pushed to the background into the foreground . No need to implement CTR-Z
- ; **Sequential execution** of commands (up to 5 commands) the argc of each command should be >=1 and <=5 as per Rule 2
  Ex: shell24$ ls -l -t ;date ; ex1 ;

You are NOT required to combined special characters : ex $ ex1 & ex2 >> output.txt

Note:

- **You must include comments throughout the program reasonably explaining the working of the code.**
- You have to use **fork() and exec() along with other pertinent system calls** to run commands from minishell
- Appropriate **error messages** must be displayed by the program based on the specifications.

Submission Instructions:

Plagiarism Detection Tool: MOSS

You need to submit the following:
1. shell24_fname_lname_SID.c
2. Zoom/Google Drive recording link explaining the following (not more than 15 minutes)

- Overall working of the code and various modules (around 8-9 minutes)
- Execution of the code under various inputs/conditions as per the requirements of the assignment (around 6-7 minutes)
- Other form of links/MP4 files will NOT be acceptable.
- Include the link in the COMMENTS section.