2022

Lonnie Operating System User Manual

VICTOR ATASIE

User's Manual

Table of Contents

User's Manual

Overview of the OS

This program simulates an operating system and provides a shell to allow the user run commands that can perform a few basic functions like, show the version or the simulator, list the files in the directory but also perform more complex functions like creating and deleting system processes.

Summary of Commands

The available shell commands in this operating system are divided into two major sections. Shell level functions and Kernel level functions. All commands are read through the shell as text input from user.

Detailed Description of Each Command

Format-

- *Name*
    - *Description*
    - *Syntax*
    - *Example*
    - *Possible Errors*

1. alias

- allow the user to change current command name
- alias [current_command_name]=[new_command_name]
- example: alias history=hist
- possible errors
    - the user enters either a wrong command name
    - the user enters either an empty new command name
    - no arguments are used with the command

2. all-pcbs

- display information for all PCBS in all queues, in order of PID.
- syntax: all-pcbs
- example: all-pcbs
- possible errors
    - the user misspells the command

3. blocked-q

- display all blocked PCBs
- syntax: blocked-q
- example: blocked-q
- possible errors –

    - the user misspells the command

4. block-pcb
    - allow the user to block a specific process
    - block-pcb –-id=[process id]
    - example: alias block-pcb –-id=3
    - possible errors –
        - the user enters an invalid process id
        - the user enters an id of a process already in the blocked queue
        - no arguments are used with the command
5. clear
    - allow the user to clear the shell
    - syntax: clear
    - example: clear
    - possible errors
        - the user misspells the command
6. delete-pcb
    - allow the user to delete a system process
    - syntax: delete-pcb –-id= [process id]
    - example: delete-pcb –-id=4
    - possible errors –
        - the user misspells the command
        - no arguments are used with the command
        - the id provided doesn't exist or is an operating system process which can't be blocked.
7. emoji
    - allow the user to change current shell emoji
    - emoji [new emoji]
    - example: emoji 🎂
    - possible errors –
        - the user misspells the command
        - no emoji is provided
8. exit
    - allow the user to stop the program (exit the shell)

- syntax: exit
- example: exit
- possible errors –
    - the user misspells the command

9. help

- displays a list of all commands
- syntax: help
- example: help
- possible errors –
    - the user misspells the command

10. history

- displays the last ten commands entered by the user
- syntax: history
- example: history
- possible errors –
    - the user misspells the command

11. ls

- displays all the files in the current directory
- syntax: ls
- example: ls
- possible errors –
    - the user misspells the command

12. new-pcb

- allow the user to create new process
- syntax: new-pcb –-id=[number] –-memory=[number] -–type=[mixed or interactive or cpu]
- example: new-pcb –-id=[20] --type=cpu –-memory=20
- possible errors –
    - the user enters either a wrong command name
    - the user doesn’t provide all the required arguments
    - no arguments are used with the command

13. generate-pcbs

- allow the user to generate a given number of processes

- syntax: generate-pcbs –-num=[number of processes to create]
- example: generate-pcbs –-num=20
- possible errors –
    - the user enters an invalid number or data type for the -–num argument
    - the user misspells the command or one of the arguments
    - no arguments are used with the command

14. ready-q

- displays a list of all the processes in the ready-q in order
- syntax: ready-q
- example: ready-q
- possible errors –
    - the user enters either a wrong command name
    - the user enters either an empty new command name
    - no arguments are used with the command

15. cpu

- allow the user to run cpu simulation(s)
- syntax: cpu --type=[fifo or round-robin or mlfq or all] –-quantum=[number] --queues=[number]
- example: cpu –-type=mlfq –-queues=10
- possible errors –
    - the user enters either a wrong command name
    - no arguments are used with the command or invalid arguments were used
    - the user misspells the command

16. script

- allow the user to run commands from the contents of a file
- syntax: script [filename.extension]
- example: script text.txt
- possible errors –
    - the user enters an invalid file name

    - no arguments are used with the command
    - the user misspells the command

17. setdate

- allow the user to set the current date of the system
- syntax: setdate [date]
- example: setdate 10/10/2020
- possible errors –
    - the user enters an invalid date string
    - no arguments are used with the command
    - the user misspells the command

18. show-pcb

- show the details of a specified pcb
- syntax: show-pcb –-d=[process id]
- example: show-pcb --id=2
- possible errors
    - the user enters an invalid process id or process with provided id doesn't exist
    - no argument is used with the command

19. unblock-pcb

- allow the user to change current command name
- syntax: unblock-pcb –-id=[blocked process id]
- example: unblock-pcb --id=20
- possible errors –
    - the user enters an invalid process id or process with provided id not blocked
    - no argument is used with the command

20. version

- allow the user to change current command name
- syntax: version
- example: version
- possible errors

- the user misspells the command

Summary of Error Messages

All error messages describe what went wrong and ask the user to either ask the user to enter the 'help' command, if the user's input did not contain a valid command, or '[COMMAND] --help' if a valid command was entered with invalid syntax or arguments.