INDEX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr No | Assignment Name | Date | Marks | Signature |
| 1. | Create a file with hole in it. |  |  |  |
| 2. | Take multiple files as Command Line Arguments and print their inode number. |  |  |  |
| 3. | Write a C program to find file properties such as inode number, number of hard link, File permissions, File size, File access and modification time and so on of a given file using stat() system call |  |  |  |
| 4. | Print the type of file where file name accepted through Command Line |  |  |  |
| 5. | Write a C program to find whether a given file is present in current directory or not. |  |  |  |
| 6. | Write a C program that a string as an argument and return all the files that begins with that name in the current directory. For example > ./a.out foo will return all file names that begins with foo |  |  |  |
| 7. | Read the current directory and display the name of the files, no of files in current directory |  |  |  |
| 8. | Write a C program which receives file names as command line arguments and display SPPU M.Sc. Computer Science Syllabus 2023-24 18 those filenames in ascending order according to their sizes. I) (e.g $ a.out a.txt b.txt c.txt, …) |  |  |  |
| 9. | Display all the files from current directory which are created in particular month |  |  |  |
| 10. | Display all the files from current directory whose size is greater that n Bytes Where n is accept from user. |  |  |  |
| 11. | Write a C Program that demonstrates redirection of standard output to a file. |  |  |  |
| 12. | Write a C program that will only list all subdirectories in alphabetical order from current directory. |  |  |  |
| 13. | Write a C program that redirects standard output to a file output.txt. (use of dup and open system call). |  |  |  |
| 14. | Write a C program to Identify the type (Directory, character device, Block device, Regular file, FIFO or pipe, symbolic link or socket) of given file using stat() system call. |  |  |  |
| 15. | Generate parent process to write unnamed pipe and will read from it |  |  |  |
| 16. | Handle the two-way communication between parent and child processes using pipe. |  |  |  |
| 17. | Demonstrate the use of atexit() function. |  |  |  |
| 18. | Write a C program to demonstrates the different behaviour that can be seen with automatic, global, register, static and volatile variables (Use setjmp() and longjmp() system call). |  |  |  |
| 19. | Implement the following unix/linux command (use fork, pipe and exec system call) ls –l | wc –l |  |  |  |
| 20. | Write a C program to create „n‟ child processes. When all „n‟ child processes terminates, Display total cumulative time children spent in user and kernel mode. |  |  |  |
| 21. | Write a C program to create an unnamed pipe. The child process will write following three messages to pipe and parent process display it.  Message1 = “Hello World”  Message2 = “Hello SPPU” Message3 = “Linux is Funny” |  |  |  |
| 22. | Write a C program to get and set the resource limits such as files, memory associated with a process. |  |  |  |
| 23. | Write a program that illustrates how to execute two commands concurrently with a pipe. |  |  |  |
| 24. | Write a C program that print the exit status of a terminated child process. |  |  |  |
| 25. | Write a C program that catches the ctrl-c (SIGINT) signal for the first time and display the appropriate message and exits on pressing ctrl-c again. |  |  |  |
| 26. | Write a C program which creates a child process and child process catches a signal SIGHUP, SIGINT and SIGQUIT. The Parent process send a SIGHUP or SIGINT signal after every 3 seconds, at the end of 15 second parent send SIGQUIT signal to child and child terminates by displaying message "My Papa has Killed me!!!”. |  |  |  |
| 27. | Write a C program to send SIGALRM signal by child process to parent process and parent process make a provision to catch the signal and display alarm is fired.(Use Kill, fork, signal and sleep system call). |  |  |  |
| 28. | Write a C program that illustrates suspending and resuming processes using signals. |  |  |  |
| 29. | Write a C program which create a child process which catch a signal sighup, sigint and sigquit. The Parent process send a sighup or sigint signal after every 3 seconds, at the end of 30 second parent send sigquit signal to child and child terminates my displaying message “My DADDY has Killed me!!!”. |  |  |  |
| 30. | Write a C program to implement the following unix/linux command (use fork, pipe SPPU M.Sc. Computer Science Syllabus 2023-24 19 and exec system call). Your program should block the signal Ctrl-C and Ctrl-\ signal during the execution. i. Ls –l | wc –l |  |  |  |
| 31. | Write a C program which creates a child process to run linux/ unix command or any user defined program. The parent process set the signal handler for death of child signal and Alarm signal. If a child process does not complete its execution in 5 second then parent process kills child process. |  |  |  |