1. Experiment with the set user bit.
2. Find how long the given file has been not changed.
3. Classify all the files in the directory based on the file type
4. Create a symbolic link file programmatically
5. Find all files in the directory which have been linked to another name(s)
6. Check whether a file is a symbolic file
7. Create a file with the given permissions
8. Display a file in reverse
9. Delete a file only if the user is the owner of the file
10. Delete all files with the size 0
11. Create a file of ‘n’ bytes
12. Make n directories and create a file in each of these directories
13. Execute a given command in the child process
14. Set an environment variable in the parent and check whether the child inherits it.
15. Set an environment variable in the child and check the parent gets that variable
16. Make an example to illustrate setjmp and longjmp
17. Demonstrate the wait system call – show the exit status and the exit value
18. Demonstrate input redirection
19. Demonstrate output direction
20. Demonstrate combining output and error streams
21. Demonstrate creating an orphan
22. Demonstrate creating a zombie
23. Set up a timer
24. Handle signal SIG\_INT; ignore SIG\_QUIT
25. Create a critical section; block signals SIG\_INT, SIG\_QUIT
26. Show the pending signals
27. Test what happens on forking to timer
28. Test what happens on execing to timer
29. Test what happens to blocked signals on forking
30. Test what happens to blocked signals on execing
31. Test what happens to pending signals on forking
32. Test what happens to pending signals on execing
33. Test what happens to signal disposition on forking
34. Show piping – make the parent send a string and child display the string
35. Show piping – execute ls cmd in the parent and wc command In the child
36. Create and display the attributes of msgqueue
37. Remove an existing msgqueue
38. Create and display the attributes of shared memory
39. Remove an existing shared memory
40. Create and display the attributes of semaphores
41. Remove an existing semaphores
42. Set the attributes of the semaphore array
43. Display the attributes of the semaphore array
44. Show an example of semop operation
45. Implement synchronization using semaphore
46. Implement alternation using semaphore
47. Implement client server using message queue
48. Implement an array in shared memory – make one process populate it and other finds the sum
49. Implement using signals synchronization between the parent and the child -(DECIDE the pattern)
50. Create a static library with at least 4 functions – write a client for this library
51. Create a dynamic library with at least 4 functions – write a client for this library
52. Create server and client TCP socket - make the server cube the number given by the client
53. Create server and client UDP socket - make the server cube the number given by the client