

## 703 : Advanced Operating System

<b>Unit 1</b>	<b>10%</b>
Introduction to Operating System Administration What is system administration? Preliminary Tasks Of System Administrator Managing User Logins Monitor System Activity And Security	
<b>Unit 2</b>	
UNIX Administration	
<b>2.1 Theory</b>	<b>20%</b>
Introduction to the Kernel The Structure of Processes Process Control	
<b>2.2 Administration</b>	<b>20%</b>
Rootly Powers Controlling Processes Adding New Users Periodic Processes Backups Using other archiving Programs Network Management And Debugging Security Web Hosting And Internet Services Policy And Politics Daemons	
<b>Unit 3</b>	<b>40%</b>
Linux Administration Rootly Powers Controlling Processes Adding New Users Periodic Processes Backups Using other archiving Programs Network Management And Debugging Security Web Hosting And Internet Services Policy And Politics Daemons	
<b>Unit 4</b>	<b>10%</b>
Windows Administration introduction Installing NT-Server Installing NT-Workstation Creating New users Assigning rights Creating groups Assigning rights Modifying users rights	

## **Text Books:**

### **1. For Unit 2.1**

The Design Of Unix Operating System

By: Batch Publication: Pearson Education Asia

Chapters : 2,6,7

### **2. For Unit 2.2**

Unix System Administration Hand Book

By: Evi Nemeth, Garth Snyder, Scott Seebass, Trent R. Hein

Publication : Low Price Edition (Pearson Education Asia)

Chapters : 3,4,6,9,10,20,21,22,27,28

### **3. For Unit 3**

Linux System Administration Hand Book

By: Evi Nemeth, Garth Snyder, Scott Seebass, Trent R. Hein

Edition : Low Price Edition (Pearson Education Asia)

Chapters : 3,4,6,9,10,20,21,22,28,29

## **(Practical List)**

1. Make the program that demonstrate the use of fork () system call.
2. Make program that demonstrates the use of getpid (), getppid (),getgid () function.
3. Prepare a one orphan process and also check for its parent process from process table.
4. Implements a zombie process and identify it from process table.
5. Demonstrate & discuss the wait () & sleep () system call.
6. Make four child process for a process, the parent process will not execute until the death of all four process. When the parent executes it should display all its four child PIDs.
7. Write a program that will create the maximum no. of process Unix allows.
8. How it is possible that for a same variable in to result (parent & Child) processes has a same address & different values demonstrate and discuss.
9. Make a process named Ex2 for display its process Id and its parents process Id , execute Ex1 from process Ex1.
10. Explain the following function with example for each
  - Execv ()
  - Execl ()
  - Ececvp ()
11. What is the advantage of exec () called through a fork () demonotstate it.
12. Calculate time taken by a child process for the execution
13. Write a program that display system details like.
  - (a) System name
  - (b) Node name
  - (c) Release
  - (d) Version
  - (e) Processor name
14. Write a program that display the user detail like
  - (a) Login name
  - (b) encrypted password
  - (c) userid
  - (d) groupid
  - (e) Password age
  - (f) comment
  - (g) Miscellany
  - (h) login directory
  - (i) shell
15. Cache the SIGINT signal and display appropriate message also display the corresponding signal key
16. Ignore the SIGINT signal.

17. Cache the SIGQUIT signal and display appropriate message also display the corresponding signal key.
  18. Ignore the SIGQUIT signal.
  19. Cache the SIGILL signal and display appropriate message also display the corresponding signal key.
  20. Ignore the SIGILL signal.
  21. Cache the SIGHUP signal and display appropriate message also display the corresponding signal key.
  22. Ignore the SIGHUP signal.
  23. Cache the SIGCLD signal and display appropriate message also display the corresponding signal key.
  24. Ignore the SIGCLD signal.
  25. Cache the SIGALRM signal and display appropriate message also display the corresponding signal key.
  26. Ignore the SIGALRM signal.
  27. Cache the SIGUSR1 signal and display appropriate message also display the corresponding signal key.
  28. Ignore the SIGUSR1 signal.
  29. Cache the SIGUSR2 signal and display appropriate message also display the corresponding signal key.
  30. Ignore the SIGUSR2 signal.
  31. Make a child process for a process, the process goes to sleep until the child process completes. When the child completes it should receive a signal. Cache the signal and display the message "CHILD DIED".
  32. Send a SIGINT signal from a parent process to a child process using kill(), the child process catches the signal SIGINT and should display an appropriate message.
  33. Install following types of Linux in graphical mode as well as non-graphical mode.
    - (a) Linux server 9.0
    - (b) Linux client 9.0
    - (c) Upgrade from 8.0 to 9.0
    - (d) Dual booting with either Windows 98/Windows-NT/Windows XP.
  34. Prepare following partition using appropriate command
 

Partition	Size
/boot	150 MB
/swap	300 MB
/	2000MB
  35. Configure following boot loaders.
    - (a) GRUB
    - (b) LILO
  36. Mount windows drives (formatted using FAT16/FAT32) in Linux.
  37. By default Linux 9.0 gives six logical screens expand it up to eleven.
  38. Add your host into network using graphical tool as well as non-graphically (i.e. using command).
  39. Add user using graphical tool as well as non-graphically (i.e. using command).
  40. Suppose you forgot your root password, how can you recover this problem and in which condition it is possible.
- 
37. By default Linux 9.0 gives six logical screens expand it up to eleven.
  38. Add your host into network using graphical tool as well as non-graphically (i.e. using command).
  39. Add user using graphical tool as well as non-graphically (i.e. using command).
  40. Suppose you forgot your root password, how can you recover this problem and in which condition it is possible.