DAY WISE SPLIT – SYSTEM PROGRAMMING USING UNIX

Days	Madhavi	Sikander
1	Introduction to UNIX Types of kernel UNIX kernel architecture Booting procedure	--------------------- No Lab --------------------------
2	UFS layout Kernel Data structures Introduction to system calls	Prototypes and programs associated with 1. open 2. Read 3. Write 4. Close Also relate the system calls used to kernel data structures
3	---------Monitor labs------------	Prototypes and programs associated with 1. Create 2. Lseek Cat and cp command implementation 1. Cat 2. Cat > file 3. Cat >> file 4. Cat file 5. Cp file1 file2
4	-----------Monitor labs ------------	Prototypes and programs associated with 1. Dup 2. Dup2 3. Fcntl for duplication 4. Fcntl for setting and getting flags Implementations for standard input and output redirections
5	File locking Implementation for entire file lock in WRITE – WRITE	Implementation of record and file locks for 1. WRITE – READ 2. READ – WRITE 3. READ – READ Getting the lock
6	Buffer management	Prototypes and programs associated with 1. Link, symlink and unlink 2. Stat, fstat and lstat 3. Chdir and mkdir Implementation of mv command Implementation of ls command using opendir and readdir
7	UNIX process management Process and process state diagram Process related kernel data structures	Prototypes and programs associated with 1. Wait 2. Waitpid

	Fork()
	Getpid()
	Getppid()
		3. Exec families
		4. Vfork

8	IPC – Signal Introduction to signals Handling signals Signal system call : example with SIGINT	Programs using 1. SIGTSTP 2. SIGQUIT 3. SIGFPE 4. SIGSEGV 5. SIGCHLD 6. SIGALRM Prototypes and programs associated with 1. Raise 2. Pause 3. Kill
9	IPC – Unnamed pipes Introduction Prototypes and programs associated with 1. Pipe Basic communication using read and write ends of pipes	Blocking effect Broken pipe and SIGPIPE Command implementations Two way communication using pipes
10	IPC –Named pipes Introduction Prototypes and programs associated with 1. Mknod and mkfifo Basic communication using read and write ends of pipes	Blocking effect Broken pipe and SIGPIPE Command implementations Two way communication using pipes
11	IPC – Message queue Introduction Prototypes and programs associated with 1. Msgget 2. Msgsnd 3. Msgrcv 4. Ipcs -q 5. Ipcrm	Programs illustrating 1. FIFO receiving 2. Random receiving 3. Two way communication 4. msgctl
12	IPC – Shared memory Introduction Prototypes and programs associated with 1. shmget 2. shmat 3. shmdt	Programs illustrating 1. reading and writing by two processes 2. shmctl ipcs –m and ipcrm
13	IPC – Semaphores	Sample programs for

	Introduction Prototypes and programs associated with 1. semget 2. semctl 3. semop Sample program for binary semaphore	1. mutex 2. counting 3. semctl with stat 4. synchronization of shared memory using semaphores
14	IPC – Socket Introduction TCP / IP implementation Sample program for TCP / IP implementation	Concurrent server and UDP implementation
15	Threads Introduction Prototypes and programs associated with 1. Pthread_self() 2. Pthread_create() 3. Pthread_exit() 4. Pthread_join() 5. Pthread_mutex_lock() 6. Pthread_mutex_unlock()	--------------Monitor lab ---------------------
16	Unix memory management	