GUJARAT TECHNOLOGICAL UNIVERSITY

INFORMATION TECHNOLOGY (23)

ADVANCE OPERATING SYSTEM
SUBJECT CODE: 2732305
M.E. SEM-III

Type of course: Regular

Prerequisite: Data Structures and Algorithms,
Advanced Programming (or good working knowledge of C), and
Fundamentals of Computer Systems
Fundamentals of Operating System

Rationale: NA

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks						Total Marks
				Theory Marks		Practical Marks				
L	T	P	C	ESE (E)	PA (M)	ESE (V)		PA (I)		
						ESE	OEP	PA	RP	
3	2#	2	5	70	30	20	10	10	10	150

Course Content:

Sr. No.	Content	Total Hrs	% Weightage
1	**Introduction to UNIX/Linux kernel** System Structure, Architecture of UNIX operating system	1	3
2	**Files & Directories IO** Function : open- creat- close- lseek-read- write, file sharing, atomic operations, dup and dup2 function , fcntl function , ioctl function, /dev/fd stat, fstat & lstat function, file types, Set-User-ID and Set-Group-ID, file access permissions, ownership of new files and directories, access function, umask function, chmod and fchmod function, sticky bit, chown, fchown, and lchown function, file size, file truncation, file systems, link, unlink, remove, and rename functions, symbolic links, symlink and readlink functions, file times, utime, mkdir and rmdir, reading directories, chdir- fchdir and getcwd funciton, device special files. **Advanced File IO** Scatter/Gather I/O, The Event Poll Interface, Mapping Files into Memory, Advice for Normal File I/O, Synchronized, Synchronous, and Asynchronous Operations, I/O Schedulers and I/O Performance	4	12
3	**Process Environment:** Process termination, environment list, memory layout of a C program, shared libraries, memory allocation, environment variables, setjmp and longjmp, getrlimit and setrlimit	5	15

	Process Control: process identifiers, fork, vfork, exit, wait and waitpid, waitid, wait3 and wait4, race conditions, exec, changing user IDs and group IDs, interpreter files, system function, process accounting, user identification, process times **Process Relationship:** Terminal logins, network logins, process groups, sessions, controlling terminal, tcgetpgrp, tcsetpgrp, and tcgetsid functions, job control, shell execution of programs, orphaned process groups		
4	**Memory Management** The Process Address Space, Allocating Dynamic Memory, Managing Data Segment, Anonymous Memory Mappings, Advanced Memory Allocation, Debugging Memory Allocations, Stack-Based Allocations, Choosing a Memory Allocation Mechanism, Manipulating Memory, Locking Memory, Opportunistic Allocation	**4**	11
5	**Signal Handling** Signal Concepts, Basic Signal Management, Sending a Signal, Re-entrancy, Signal Sets, Blocking Signals, Advanced Signal Management, Sending a Signal with a Payload	**4**	11
6	**Introduction to Distributed Operating System** **Theoretical Foundations:** Limitations, Lamport's logical clock, Vector clock, causal ordering, global state, Cuts.	**3**	8
7	**Distributed Mutual Exclusion:** Lamport, Recart-agrawala, and Maekawa's algorithms; Suzuki-kasami broadcast algorithm, and Raymond's tree based algorithm	**3**	8
8	**Distributed Deadlock Detection:** Resource Vs. Communication deadlock, Strategies to handle deadlock, Ho-Ramamoorthy, Path-Pushing, Edge-Chasing, Diffusion Computation-based algorithms. **Agreement Protocols:** System model, Classification of agreement problems, Solutions to Byzantine agreement problems.	**6**	16
9	**Distributed File Systems:** Mechanisms for building DFSs, Design Issues **Distributed Scheduling:** Issues in Load Distribution, Components of a load distribution algorithm, Load Distribution Algorithms.	3	8
10	**Distributed Shared Memory** Algorithm for implementing DSMs, Memory Coherence ,and Coherence protocol	**3**	8

Reference Books:

1. Advanced Concepts in Operating Systems Singhal, Mukesh & N.G. Shivaratri, Tata McGraw-Hill, 1994
2. Advanced Programming in the UNIX Environment, Addison-Wesley, by Richard Stevens.
3. Linux System Programming, O'Reilly, by Robert Love.
4. Distributed Operating Systems" Pearson Education, 1998. P. K. Sinha,
5. Distributed Operating Systems – The Logical Design by A. Goscinski, AW
6. Modern Operating Systems by A. S. Tanenbaum, Pearson Education
7. The Design of the UNIX Operating System, PHI, by Maurice J. Bach

Course Outcome:

1. Discuss the various synchronization, scheduling and memory management issues.
2. Demonstrate the mutual exclusion, deadlock detection and agreement protocol of Distributed operating system.
3. Discuss the various resource management techniques for distributed system

1. Write your own dup2 function that performs the same service as the dup2 system call without calling the fcntl function. Be sure to handle errors correctly.
2. Write a utility like cp(1) that copies a file containing holes, without writing the bytes of 0 to the output file.
3. Write a C program that creates a zombie, and then call systemto execute the ps(1) command to verify that the process is a zombie.
4. Implement your own sig2str function.
5. Write a C program that creates a file and writes the integer 0 to the file. The process then creates a child, and the Parent and Child alternate incrementing the counter in the file. Each time the counter is incremented, print which process (Parent or Child) is doing the increment.
6. Write a C program that calls fork and has the child create a new session. Verify that the child becomes a process group leader and that the child no longerhas a controlling terminal.
7. Write a C function which handles all possible signals. The function should consist of a single loop that iterates once for every signal in the current signal mask (not once for every possible signal).
8. Write a C program that calls sleep (60) in an infinite loop. Every five times through the loop (every 5 minutes), fetch the current time of day and print the tm_sec field.
9. Write a C program that calls fwrite with a large buffer (a few hundred megabytes). Before calling fwrite, call alarm to schedule a signal in 1 second. In your signal handler, print that the signal was caught and return.
10. Design, develop and implement a process with a producer thread and a consumer thread which make use of bounded buffer (size can be prefixed at a suitable value) for communication. Use any suitable synchronization construct.
11. Write an interface and implements server that will sends Hello string to the client when requested.
12. Write a Client program that will invoke remote method of server implementation to get the time service and the server serves the request by sending back the date time value on server.
13. Write a program to find the length of string using thread.
14. Write a client program that will connect with the server port and sends request to the server and the server sends the echo message in reply to client using java socket programming
15. Design, develop, and execute a program to solve a system of n linear equations using Successive Over-relaxation method and n processes which use Shared Memory API.

Design based Problems (DP)/Open Ended Problem:

1. Discuss & Compare different distributed mutual exclusion algorithm. Implement theoretically.
2. Discuss & analyze agreement protocol.
3. Analyze load distribution algorithm

Major Equipment:

i Linux based Host machines (Free & Open Source Software or Open source)
ii Computers with latest hardware configuration

List of Open Source Software/learning website:

- Operating System concepts: http://nptel.iitm.ac.in/
- Linux basics: www.freeos.com/guides/lsst
- Linux basics: www.linuxcommand.org/writing_asell_scripts.
- Linux basics: www.distro.ibiblio.org/damnsmall/current/dsl-4.4.10- embedded.zip

Review Presentation (RP): The concerned faculty member shall provide the list of peer reviewed Journals and Tier-I and Tier-II Conferences relating to the subject (or relating to the area of thesis for seminar) to the students in the beginning of the semester. The same list will be uploaded on GTU website during the first two weeks of the start of the semester. Every student or a group of students shall critically study 2 papers, integrate the details and make presentation in the last two weeks of the semester. The GTU marks entry portal will allow entry of marks only after uploading of the best 3 presentations. A unique id number will be generated only after uploading the presentations. Thereafter the entry of marks will be allowed. The best 3 presentations of each college will be uploaded on GTU website.