Nirma University
Institute of Technology
CSE Department
Course Policy
B. Tech. in Computer Engineering
Semester: IV, Academic Year: 2020-2021, Term: EVEN

Course Code & Name	2CS403 Operating System
Credit Details	4
Course Co-ordinator	Prof. Vishal Parikh
Contact No. & Email	vishalparikh@nirmauni.ac.in
Office	N block
Visiting Hours	9:00 am – 4:00 pm
Course Faculty	Prof. Parita Oza Prof. Chandan Trivedi Prof. Deepti Saraswat Prof. Umesh Bodkhe
Contact No. & Email	parita.prajapati@nirmauni.ac.in chandan.trivedi@nirmauni.ac.in deepti.saraswat@nirmauni.ac.in umesh.bodkhe@nirmauni.ac.in

Introduction to Course:

As computers became more complex, it became necessary to have a layer between the programs and the hardware that could administer the interactions of one with the other. The fundamental job of an operating system has always been the same: it enables applications and their users to interact with the various hardware components including the hard disc, network, graphics card and memory. This course covers the classical internal algorithms and structures of operating systems, including CPU scheduling, memory management, and device management. It also covers the unifying concept of the operating system as a collection of cooperating sequential processes.

Pre-Requisites: Computer Fundamentals

Course Website:- https://sites.google.com/nirmauni.ac.in/2cs403-operating-system

Course Learning Outcomes (CLO):

At the end of the course, students will be able to -

1. illustrate basic components of operating systems
2. comprehend the mechanism of operating Systems to handle processes, memory and file management
3. demonstrate competence in recognizing and using operating system features

.Syllabus	Teaching Hours
Unit I **Overview of Computer System and Operating System:** Elements of computer system, operating system objectives and functions, evolution of operating systems	3

Unit II **18**

Process Description and Control: Process states, process description, process control, process management, Uniprocessor scheduling, multiprocessor and real-time scheduling, case study

Unit III **3**

Threads: Processes And Threads, Symmetric Multiprocessing, Micro kernels

Unit IV **8**

Concurrency: Mutual exclusion and synchronization, deadlock and starvation, case study

Unit V **10**

Memory Management and Virtual Memory: Memory management requirements, partitioning, paging, segmentation, virtual memory, case study

Unit VI **3**

I/O Management and Files: I/O devices, organization of I/O functions, OS design issues, I/O buffering, disk scheduling, disk cache , file management, security aspects in OS, case study

Self-study:

As per the syllabus notification, 10% of syllabus is to be identified as the self-study component. The aim of this exercise is to impart the life-long learning skills among the budding professionals. Following topics are identified as the self – learning component:

- Multiprocessor
- Real-time scheduling

Laboratory work details: (List of Experiments, Schedule, assessment policy)

Laboratory work will be based on above syllabus with minimum 10 experiments to be incorporated.

Total Marks	**Continuous Evaluation**			**Semester End Evaluation**	
100 marks	No. of experiments	Max. Marks	Weightage	Max Marks	Weightage
	10	100	75%	25	25%

During each session, faculty member will introduce the objective and methodology of the practical to the students. Students are required to design the logic and implement the same. The implementation is to be done using linux shell. After implementation, student will submit the file (written neatly in tutorial pages) to the concerned faculty member.

List of Practical

Sr. No	Definition of Practical	Hours	CLO
1	a) Getting acquaintance with basic UNIX commands. b) Getting acquaintance with UNIX filters. **(Appendix - 1)**	4	3
2	Write a shell script for performing the functions of a basic calculator. (Using decision making and case control structure).	2	3
3	a) Write a shell script to compare the contents of two files. b) Write a shell script to generate all the combinations of 1, 2 and 3.	2	3
4	(a)Write a shell script to keep on accepting lines of text and write the text into a data file until the user inputs "end". The script should count the number of lines input and display them. (b)Write a shell script to print the reverse of an input number.	2	3
5	(a) Write a shell script which imitates head command. (b) Write a shell script which imitates tail command.	2	3
6	a) Write a shell script to generate the series of number multiply by 2. b) Write a shell script to concatenate all given file into a single file.	2	3
7	Write a shell script for implementing directory management.	4	3
8	Write a shell script for performing basic functions related to DBMS.	4	3
9	Write a C program to implement a system call using the fork() and Exec() function.	4	3
10	Write a C program to implement grep command.	4	3
11*	Write a C Program to implement CPU scheduling algorithm. Display average waiting time and average TAT.	4	2,3
12*	Write a C Program to implement memory allocation algorithms.	4	2,3
13*	A) Create a bootable pendrive for installing your favourite OS into system. B) Create an ISO file from your bootable pendrive and use this image file to boot in hypervisor(for eg. VMware).	2	1,3
14*	Write a C program to implement banker's algorithm to demonstrate resource allocation amongst processes while avoiding deadlock.	4	2,3
15*	Demonstrate the concept of thread using pthread library.	2	1,3

***Optional**

Appendix - 1

- Unix System Organization
- Types of shells
- Commands

who	who am i	touch	cat
cp	rm	mv	ls
ln	chmod	umask	pwd
mkdir	rmdir	cd	bc
bc -l	expr	factor	logname

uname	tty	date	df
du	ulimit	cal	wc
sort	cut	grep	awk
head	pg	more	tail
pipe (|)	tee	ps	kill
nice	read	echo	I/O redirection

- **Shell programing**
  - Shell variables
  - Arithmetic in shell script
  - Decision making instructions
  - File test
  - String test
  - Numerical test
  - Logical operators
  - Case control structures
  - Loop control structures

Suggested Readings:

1. William Stallings, Operating Systems, PHI.
2. Silberschiltz, Galvin and Greg Gange, Operating System, Willey India.
3. Sumitabha Das, Unix Concepts and Applications, TMH Publications.
4. Yashvant Kanetkar, Shell Programming, BPB.
5. A.S.Tannenbaum, Modern Operating Systems, TMH Publications.
6. Kernighan, the UNIX Programming Environment, Pearson
7. Maurice Bach, The Unix Operating System, Prentice Hall

Lesson Plan

Sr. No.	Topics	Hours	CLOs	Application
1	**Unit I** **Overview of Computer System and Operating System:** • Elements of computer system • Operating system objectives and functions • Evolution of operating systems	**[3]** 1 1 1	1	Virtualization in distributed systems
2	**Unit II** **Process Description and Control:** • Process states • Process description • process control • Process management • Uniprocessor scheduling	**[18]** 2 2 2 4 4 4	2,3	Developing Schedulers for cloud computing environments / Distributed Systems

	- Multiprocessor and Real-time scheduling (Self Study)			
3	**Unit III** **Threads:** - Processes And Threads - Symmetric Multiprocessing - Micro kernels	**[3]** 1 1 1	1	Remote Method Communication Like RPC/RMI
4	**Unit IV** **Concurrency:** - Mutual exclusion and synchronization - Deadlock and starvation	**[8]** 4 4	2,3	Distributed deadlock
5	**Unit V** **Memory Management and Virtual Memory:** - Memory management requirements - Partitioning - Paging - Segmentation - Virtual memory	**[10]** 2 2 2 2 2	2,3	Virtualization at cloud computing
6	**Unit VI** **I/O Management and Files:** - I/O devices, Organization of I/O function & OS design issues - I/O buffering & Disk scheduling - Disk cache & File management	**[3]** 1 1 1	2	Disk Management, Network File Management
	Total	45		

In order to maintain consistency among all divisions studying the course, it is required to have a uniform teaching-learning-assessment-evaluation policy across all the divisions. The Session Learning Outcomes (SLOs) are as under:

Mapping of Session Learning Outcomes (SLO) with Course Learning Outcomes (CLO)

Session No.	**Session Learning Outcomes:** **After successful completion of the session, student will be able to**	**CLO**
1.	understand need of operating system and its types	1
2.	acquire knowledge about evolution of OS	1
3.	understand steps involved in function calls	1

4.	Understand concept of process and program	2,3
5.	Understand concept of process and program	2,3
6.	learn about how processes are stored in main memory	2,3
7.	learn about how processes are stored in main memory	2,3
8.	categorize different levels of scheduler and their role in process state diagram	2,3
9.	categorize different levels of scheduler and their role in process state diagram	2,3
10.	learn working of process scheduling algorithm	2,3
11.	learn working of process scheduling algorithm	2,3
12.	learn working of process scheduling algorithm	2,3
13.	compare working of process scheduling algorithm	2,3
14.	compare working of process scheduling algorithm	2,3
15.	learn about different characteristics of different process scheduling algorithm used in real world	2,3
16.	learn about different characteristics of different process scheduling algorithm used in real world	2,3
17.	learn about multiprocessor	2,3
18.	learn about multiprocessor	2,3
19.	learn about real time scheduling	2,3
20.	learn about real time scheduling	2,3
21.	learn about real time scheduling	2,3
22.	identify need of thread and its working	2,3
23.	identify need of thread and its working	2,3
24.	compare user level and kernel level threads	2,3
25.	infer problems that occur due to race condition	2,3
26.	compare different solutions to race condition	2,3
27.	compare different solutions to race condition	2,3
28.	learn classical IPC problem and its possible solution	2,3
29.	learn classical IPC problem and its possible solution	2,3
30.	understand characteristic of deadlock problems	2,3
31.	learn how banker's algorithm helps to avoid deadlock	2,3
32.	Learn how banker's algorithm helps to detect deadlock and recover from it.	2,3
33.	understand different memory management techniques	2,3
34.	compare different partition allocation techniques	2,3
35.	correlate MVT importance over MFT	2,3
36.	correlate MVT importance over MFT	2,3
37.	conceptualize need of paging	2,3
38.	conceptualize need of paging	2,3
39.	compare different page replacement algorithm	2,3
40.	compare different page replacement algorithm	2,3
41.	recognize the need of segmentation	2,3

42.	recognize the need of segmentation with paging	2,3
43.	learn about different I/O devices and its need	2
44.	Learn about disk arm scheduling	2
45.	learn about file and directory structure, understand file system implementation	2

Course Assessment Schemes

Assessment scheme	**CE**			**LPW**		**SEE**
Component weightage	0.4			0.2		0.4
	Class Test 30%	Sessional Exam 40%	Innovative Assignment 30%	Continuous Evaluation 75%	Viva Voce 25%	

It is mandatory to clear each component with minimum 40%.

Teaching-learning methodology:

- Lectures: Use of Black board, PPT, Discussion, Case Studies etc.
- Laboratory: The teaching-learning methodology and assessment policy is as described in the section of Laboratory Details.

Active learning techniques (Mention the proposed)

- Flipped Class-room
- Muddiest Points
- Team quizzes
- Just in time teaching

Types of Special/Innovative Assignments, Term Papers, mini Projects etc.

As a part of Innovative Assignment, students are required to make a small project demonstrating the concepts of an operating systems they have studied.

Course Material:

All relevant information covering various aspects of the course is put on the website at the below mentioned link:

Course Outcome Attainment:

Following means will be used to assess attainment of course learning outcomes:

- Use of formal evaluation components of continuous evaluation, laboratory work, semester end examination
- Informal feedback during course conduction

Academic Integrity Statement:

- Students are expected to carry out assigned work under Continuous Evaluation (CE) component and LPW component. Copying in any form is not acceptable and will invite strict disciplinary action. Evaluation of corresponding component will be affected proportionately in such cases. Academic integrity is expected from students in all components of course assessment