

## NIRMA UNIVERSITY

<b>Institute:</b>	Institute of Technology
<b>Name of Programme:</b>	B.Tech. Computer Science and Engineering
<b>Course Code:</b>	2CSDE82
<b>Course Title:</b>	Real Time Operating Systems
<b>Course Type:</b>	Departmental Elective
<b>Year of Introduction:</b>	2021-22

### Credit Scheme

L	T	Practical Component				C
		LPW	PW	W	S	
3	0	2	-	-	-	4

### Course Learning Outcomes (CLO):

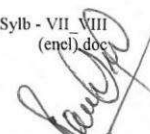
At the end of the course, students will be able to –

1. summarize the characteristics of a real-time system
2. apply scheduling concepts to real time applications
3. examine the causes of fault occurrence in real time operating systems
4. evaluate intricacies of real time databases

### Syllabus:

**Total Teaching hours: 45**

Unit	Syllabus	Teaching hours
Unit-I	<b>Introduction:</b> Introduction to Real Time Systems, Structure of a Real Time System, Hard Real Time Systems and Soft Real Time Systems, Issues in Real Time Computing, Real Time Applications: Industrial Applications, Telecommunications Applications, Internet and Multimedia Applications, Digital Control, High Level Controls, Signal Processing etc.	08
Unit-II	<b>Real Time Task Scheduling:</b> Task Scheduling, Clock-Driven Scheduling, Hybrid Schedulers, Event-Driven Scheduling, Earliest Deadline First (EDF) Scheduling, Rate Monotonic Algorithm (RMA), Issues associated with RMA	10
Unit-III	<b>Handling Resource Sharing and Dependencies Among Real-Time Tasks:</b> Resource Sharing Among Real – Time Tasks, Priority Inversion, Priority Inheritance Protocol, Highest Locker Protocol, Priority Ceiling Protocol, Types of Priority Inversions Under PCP, Important Features of PCP, Issues in Resource Sharing Protocols, Handling Task Dependencies.	10



Unit-IV	<b>Failure Management and Recovery:</b> Causes of Failure, Fault Types, Fault Detection, Fault and Error Containment, Hardware and Software Redundancy, Time Redundancy, Information Redundancy. Data Diversity, Integrated Failure Handling.	10
Unit-V	<b>Commercial Real-Time Operating Systems &amp; Real Time Database:</b> Features, UNIX as Real-Time Operating System, POSIX, Real time Vs General Purpose Databases, Main Memory Databases, Transaction priorities, Transaction Aborts, Concurrency Control Issues, Two-phase Approach to improve Predictability, Maintaining Serialization Consistency, Databases for Hard Real Time systems.	07

Self-Study: The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents

Suggested Readings/References:

1. Mall Rajib, Real Time Systems Pearson Education
2. C.M. Krishna, Kang G. Shin, Real Time Systems, McGraw Hill
3. Marilyn Wolf, Computers as Components Principles of Embedded Computing System Design, Morgan Kaufmann

Suggested List of Experiments:	Sr. No.	Practical Title	Hours
	1	To study UNIX commands and utilities like utilities like make, tar	02
	2	To use profiling commands like prof and gprof and analyse the code of your sample program	02
	3	To study process system calls:	04
		A. Write a program that shows use of fork () system call.	
		B. Write a program that explains about process tree. (Description: As fork() is called 3 times, 2 <sup>3</sup> processes will be created. That is in total we may have 8 times "Hello" message to be printed on screen.)	
		C. Write a program that creates process chain. This program creates in total four processes such that every parent has one child exactly.	
	4	To implement real task scheduling: Earliest Deadline First	04
	5	To implement real task scheduling: Rate Monotonic Algorithm	02

- |    |  |    |
|----|--|----|
| 6  | To write a program that demonstrates the creation of shared memory   | 04 |
| 7  | To write a program that calculates the fraction of time that a process is running on a processor while performing a computation                          | 02 |
| 8  | To write an application that creates two tasks of the same priority and sets the time slice period to illustrate time slicing                            | 04 |
| 9  | To write a program for sending messages to mailbox by one task and reading the message from mailbox by another task                                      | 02 |
| 10 | To install ARM cross compiler tool chain and write a sample program in C and compile it for ARM target and to study the assembly code and write comments | 04 |

Suggested Case List: -NA-