**ASSIGNMENT-1**

Module- 1: Basics of Real-Time Operating Systems

1. Define a real-time operating system (RTOS) and explain its key characteristics.
2. Differentiate between hard real-time and soft real-time systems.
3. What are the primary goals of a real-time operating system?
4. List and explain the main components of a real-time operating system.
5. What is the difference between a real-time task and a non-real-time task?
6. Describe the challenges faced in scheduling real-time tasks.
7. Discuss the role of interrupt handlers in real-time operating systems.
8. Explain the concept of preemption and its importance in real-time systems.
9. Compare and contrast periodic and aperiodic tasks in real-time systems.
10. What are the different types of real-time scheduling algorithms? Explain each briefly.

Module-2: Task Scheduling and Synchronization

1. Discuss the significance of priority-based scheduling in real-time operating systems.
2. Explain the Rate Monotonic Scheduling (RMS) algorithm and its limitations.
3. What is the Earliest Deadline First (EDF) scheduling algorithm? How does it work?
4. Describe the concept of priority inversion and its impact on real-time systems.
5. What is resource sharing in real-time systems? Discuss the challenges associated with it.
6. Explain the concept of mutual exclusion and its importance in real-time systems.
7. Compare and contrast semaphores and mutexes in terms of usage and behavior.
8. Describe the reader-writer problem and discuss possible solutions in real-time systems.
9. What is a deadlock in real-time systems? Explain the necessary conditions for deadlock occurrence.
10. Discuss the concept of priority inheritance and its role in preventing priority inversions.

**ASSIGNMENT-2**

Module- 3: Memory Management and Communication

1. Describe the memory requirements and challenges in real-time operating systems.
2. Explain the concept of memory partitioning and its advantages and disadvantages.
3. Discuss the role of virtual memory in real-time systems and its impact on performance.
4. What is inter-process communication (IPC)? Explain its significance in real-time systems.
5. Compare and contrast message queues and shared memory as IPC mechanisms.
6. Explain the concept of remote procedure call (RPC) and its usage in real-time systems.
7. Discuss the challenges and strategies for achieving reliable communication in real-time systems.
8. Describe the concept of time-triggered communication and its benefits in real-time systems.
9. What is the role of network protocols in real-time distributed systems?
10. Explain the concept of fault tolerance and its importance in real-time systems.

Module- 4: Real-Time Operating System Design and Case Studies

1. Discuss the key factors to consider in designing a real-time operating system.
2. Explain the concept of a task control block (TCB) and its role in real-time systems.
3. Describe the concept of a scheduler and its types in real-time operating systems.
4. What is the role of a context switch in real-time systems? Explain its impact on performance.
5. Discuss the challenges and techniques for reducing interrupt latency in real-time systems.
6. Explain the concept of time-triggered scheduling and its applications.

**ASSIGNMENT-3**

1. Describe the Real-Time Executive for Multiprocessor Systems (RTEMS) and its features.
2. Discuss the use of real-time operating systems in automotive systems.
3. Explain the design considerations and challenges in real-time operating systems for aerospace applications.
4. Describe the Real-Time Linux operating system and its use cases.
5. Module- 5: Real-Time Operating System Testing and Verification
6. What are the challenges in testing and verification of real-time operating systems?
7. Discuss the concept of worst-case execution time (WCET) analysis in real-time systems.
8. Explain the importance of simulation and emulation in testing real-time operating systems.
9. Discuss the concept of formal verification and its role in ensuring system correctness.
10. Describe the challenges and techniques for testing real-time operating system kernels.
11. Explain the concept of system integration testing in real-time operating systems.
12. What is fault injection testing? Discuss its significance in real-time systems.
13. Describe the concept of model-based testing and its usage in real-time systems.
14. Discuss the challenges and techniques for testing real-time operating system drivers.
15. Explain the concept of safety-critical systems and the importance of testing in such systems.