**UNIT 2**

**PROCESS SYNCHRONIZATION,CPU SCHEDULING AND DEADLOCK**

**PART A**

**MULTIPLE CHOICE QUESTIONS:**

1. ***Which process can be affected by other processes executing in the system?(Pgno:209) [L1]  
    a) cooperating process  
    b) child process  
    c) parent process  
    d) init process***

***Answer : a***

1. ***Peterson solution is restricted to ----------- process that alternate execution between their critical section and remainder sections.(Pgno:206) [L2]***

***a)one  
 b) two  
 c) three  
 d) four***

***Answer : b***

1. ***When several processes access the same data concurrently and the outcome of the execution depends on the particular order in which the access takes place, is called?(Pgno:211) [L1]  
    a) dynamic condition  
    b) race condition  
    c) essential condition  
    d) critical condition***

***Answer : b***

1. ***Mutex is short for ------(Pgno:206) [L1]***

***a)Mutual Excluded  
 b) mutually explained  
 c) mutual exclusion  
 d) mutual exception***

***Answer : c***

1. ***If a process is executing in its critical section, then no other processes can be executing in their critical section. This condition is called?(Pgno:207) [L1]  
    a) mutual exclusion  
    b) critical exclusion  
    c) synchronous exclusion  
    d) asynchronous exclusion***

***Answer : a***

1. ***Which one of the following is a synchronization tool?(Pgno:214) [1]  
    a) thread  
    b) pipe  
    c) semaphore  
    d) socket***

***Answer : c***

1. ***A semaphore is a shared integer variable \_\_\_\_\_\_\_\_\_\_(Pgno:214) [L1]  
    a) that can not drop below zero  
    b) that can not be more than zero  
    c) that can not drop below one  
    d) that can not be more than one***

***Answer : a***

1. ***Spinlocks are intended to provide-------(Pgno:213) [L1]***

***a)mutual exclusion  
 b) bounded waiting  
 c) Aging  
 d) Progress***

***Answer : b***

1. ***Mutual exclusion can be provided by the \_\_\_\_\_\_\_\_\_\_(Pgno:213) [L1]  
    a) mutex locks  
    b) binary semaphores  
    c) both mutex locks and binary semaphores  
    d) binary lock***

***Answer : c***

1. ***When high priority task is indirectly preempted by medium priority task effectively inverting the relative priority of the two tasks, the scenario is called \_\_\_\_\_\_\_\_\_\_(pg-no:216) [L1]  
    a) priority inversion  
    b) priority removal  
    c) priority exchange  
    d) priority modification***

***Answer : a***

1. ***Process synchronization can be done on \_\_\_\_\_\_\_\_\_\_(Pgno:209) [1]  
    a) hardware level  
    b) software level  
    c) both hardware and software level  
    d) Process Level***

***Answer : c***

1. ***Semaphore is a/an \_\_\_\_\_\_\_ to solve the critical section problem.(Pgno:215) [L1]  
    a) hardware for a system  
    b) special program for a system  
    c) integer variable  
    d) Software for a system***

***Answer : c***

1. ***The wait operation of the semaphore basically works on the basic \_\_\_\_\_\_\_ system call.(Pgno:216) [L2]  
    a) stop()  
    b) block()  
    c) hold()  
    d) wait()***

***Answer : b***

1. ***The signal operation of the semaphore basically works on the basic \_\_\_\_\_\_\_ system call.(Pgno:215) [L2]  
    a) continue()  
    b) wakeup()  
    c) getup()  
    d) start()***

***Answer : b***

1. ***What are the two kinds of semaphores?(Pgno:214) [L1]  
    a) mutex & counting  
    b) binary & counting  
    c) counting & decimal  
    d) decimal & binary***

***Answer : b***

1. ***At a particular time of computation the value of a counting semaphore is 7.Then 20 P operations and 15 V operations were completed on this semaphore. The resulting value of the semaphore is? [L3]  
    a) 42  
    b) 2  
    c) 7  
    d) 12***

***Answer : b***

1. ***A binary semaphore is a semaphore with integer values \_\_\_\_\_\_\_\_\_\_\_\_(Pgno:214) [L2]  
    a) 1  
    b) -1  
    c) 0.8  
    d) 0.5***

***Answer : a***

1. ***Spinlocks are intended to provide \_\_\_\_\_\_\_\_\_\_ only.(Pgno:213) [L2]  
    a) Mutual Exclusion  
    b) Bounded Waiting  
    c) Aging  
    d) Progress***

***Answer : b***

1. ***The bounded buffer problem is also known as \_\_\_\_\_\_\_\_\_\_\_\_(pg-219) [L1]  
    a) Readers – Writers problem  
    b) Dining – Philosophers problem  
    c) Producer – Consumer problem***

***d) Critical Section Problem***

***Answer : c***

1. ***In the bounded buffer problem \_\_\_\_\_\_\_\_\_\_\_\_(Pgno:219) [L2]  
    a) there is only one buffer  
    b) there are n buffers ( n being greater than one but finite)  
    c) there are infinite buffers  
    d) the buffer size is bounded***

***Answer : b***

1. ***The dining – philosophers problem will occur in case of \_\_\_\_\_\_\_\_\_\_\_\_(Pgno:222) [L2]  
    a) 5 philosophers and 5 chopsticks  
    b) 4 philosophers and 5 chopsticks  
    c) 3 philosophers and 5 chopsticks  
    d) 6 philosophers and 5 chopsticks***

***Answer : a***

1. ***To ensure difficulties do not arise in the readers – writers problem \_\_\_\_\_\_\_ are given exclusive access to the shared object.(Pgno:220) [L1]  
    a) readers  
    b) writers  
    c) readers and writers  
    d) either reader or writers***

***Answer : b***

1. ***A deadlock free solution to the dining philosophers problem \_\_\_\_\_\_\_\_\_\_\_\_(Pgno:223) [L2]  
    a) necessarily eliminates the possibility of starvation  
    b) does not necessarily eliminate the possibility of starvation  
    c) eliminates any possibility of any kind of problem further  
    d) does not eliminates any kind of problem***

***Answer : b***

1. ***All processes share a semaphore variable mutex, initialized to 1. Each process must execute wait(mutex) before entering the critical section and signal(mutex) afterward.  
   Suppose a process executes in the following manner.(Pgno:224) [L2]***

***Wait(mutex);***

***…..***

***Critical section***

***….***

***Wait(mutex);***

***a deadlock will occur  
 b) processes will starve to enter critical section  
 c) several processes maybe executing in their critical section  
 d) Critical section probem may occur***

***Answer : a***

1. ***In the bounded buffer problem, there are the empty and full semaphores that \_\_\_\_\_\_\_\_\_\_\_\_(Pgno:219) [L2]  
    a) count the number of empty and full buffers  
    b) count the number of empty and full memory spaces  
    c) count the number of empty and full queues  
    d) count the number of empty and full process***

***Answer : a***

***26. Which module gives control of the CPU to the process selected by the short-term scheduler? (Pgno: 265) [L1]***

***a) dispatcher***

***b) interrupt***

***c) scheduler***

***d) context switch***

***Answer: a***

***27. The processes that are residing in main memory and are ready and waiting to execute are kept on a list called \_\_\_\_\_\_\_\_\_\_\_\_\_ (Pgno: 267) [L1]***

***a) job queue***

***b) ready queue***

***c) execution queue***

***d) process queue***

***Answer: b***

***28. The interval from the time of submission of a process to the time of completion is termed as \_\_\_\_\_\_\_\_\_\_\_\_( Pgno: 265) [L1]***

***a) waiting time***

***b) turnaround time***

***c) response time***

***d) throughput***

***Answer: b***

***29. Which scheduling algorithm allocates the CPU first to the process that requests the CPU first? (Pgno: 266) [L2]***

***a) first-come, first-served scheduling***

***b) shortest job scheduling***

***c) priority scheduling***

***d) Round robin scheduling***

***Answer: a***

***30. In priority scheduling algorithm \_\_\_\_\_\_\_\_\_\_\_\_ (Pgno: 270) [L1]***

***a) CPU is allocated to the process with highest priority***

***b) CPU is allocated to the process with lowest priority***

***c) Equal priority processes can not be scheduled***

***d)processes are allocated by order***

***Answer: a***

***31. In priority scheduling algorithm, when a process arrives at the ready queue, its priority is compared with the priority of \_\_\_\_\_\_\_\_\_\_\_\_ (Pgno: 270) [L2]***

***a) all process***

***b) currently running process***

***c) parent process***

***d) init process***

***Answer: b***

***32. Which algorithm is defined in Time quantum? (Pgno: 271) [L1]***

***a) shortest job scheduling algorithm***

***b) round robin scheduling algorithm***

***c) priority scheduling algorithm***

***d) multilevel queue scheduling algorithm***

***Answer: b***

***33. Process are classified into different groups in \_\_\_\_\_\_\_\_\_\_\_\_ (Pgno: 273) [L1]***

***a) shortest job scheduling algorithm***

***b) round robin scheduling algorithm***

***c) priority scheduling algorithm***

***d) multilevel queue scheduling algorithm***

***Answer: d***

***34. In multilevel feedback scheduling algorithm \_\_\_\_\_\_\_\_\_\_\_\_ (Pgno: 275) [L1]***

***a) a process can move to a different classified ready queue***

***b) classification of ready queue is permanent***

***c) processes are not classified into groups***

***d) processes are classified into groups***

***Answer: a***

***35. Which one of the following cannot be scheduled by the kernel? (Pgno: 277) [L2]***

***a) kernel level thread***

***b) User level thread***

***c) process***

***d) process as well as kernel***

***Answer: b***

***36. With round robin scheduling algorithm in a time shared system \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 272) [L2]  
a) using very large time slices converts it into First come First served scheduling algorithm  
b) using very small time slices converts it into First come First served scheduling algorithm  
c) using extremely small time slices increases performance  
d) using very small time slices converts it into Shortest Job First algorithm***

***Answer: a***

***37. What is FIFO algorithm? (Pgno: 266) [L1]  
a) first executes the job that came in last in the queue  
b) first executes the job that came in first in the queue  
c) first executes the job that needs minimal processor  
d) first executes the job that has maximum processor needs***

***Answer: b***

***38. There are 10 different processes running on a workstation. Idle processes are waiting for***

***an input event in the input queue. Busy processes are scheduled with the Round-Robin***

***time sharing method. Which out of the following quantum times is the best value for***

***small response times, if the processes have a short runtime, e.g. less than 10ms? (Pgno: 272) [L2]  
a) tQ = 15ms  
b) tQ = 40ms  
c) tQ = 45ms  
d) tQ = 50ms***

***Answer: a***

***39. Which of the following algorithms tends to minimize the process flow time? (Pgno: 268) [L1]  
a) First come First served  
b) Shortest Job First  
c) Earliest Deadline First  
d) Longest Job First***

***Answer: b***

***40. Which of the following statements are true? (Pgno: 267,271) [L2]***

***I. Shortest remaining time first scheduling may cause starvation***

***II. Preemptive scheduling may cause starvation***

***III. Round robin is better than FCFS in terms of response time***

***a) I only  
b) I and III only  
c) II and III only  
d) I, II and III***

***Answer: d***

***41. A system is in the safe state if \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 328) [L1]  
a) the system can allocate resources to each process in some order and still avoid a deadlock  
b) there exist a safe sequence  
c) the system cannot allocate resources to each process in some order and still avoid a deadlock  
d) enters into deadlock condition***

***Answer: a***

***42. The circular wait condition can be prevented by \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 325) [L2]***

***a) defining a linear ordering of resource types***

***b) using thread***

***c) using pipes***

***d) using both thread and pipes.***

***Answer: a***

***43. Which one of the following is the deadlock avoidance algorithm? (Pgno: 330) [L2]***

***a) banker’s algorithm***

***b) round-robin algorithm***

***c) elevator algorithm***

***d) karn’s algorithm***

***Answer: a***

***44. A problem encountered in multitasking when a process is perpetually denied necessary resources is called \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 338) [L2]***

***a) deadlock***

***b) starvation***

***c) inversion***

***d) aging***

***Answer: b***

***45. To avoid deadlock \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 327) [L2]  
a) there must be a fixed number of resources to allocate  
b) resource allocation must be done only once  
c) all deadlocked processes must be aborted  
d) inversion technique can be used***

***Answer: a***

***46. The number of resources requested by a process \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 327) [L2]  
a) must always be less than the total number of resources available in the system  
b) must always be equal to the total number of resources available in the system  
c) must not exceed the total number of resources available in the system  
d) must exceed the total number of resources available in the system***

***Answer: c***

***47. The request and release of resources are \_\_\_\_\_\_\_\_\_\_\_(Pgno: 328) [L2]  
a) command line statements  
b) interrupts  
c) system calls  
d) special programs***

***Answer: c***

***48. Deadlock prevention is a set of methods \_\_\_\_\_\_\_\_\_\_\_\_(Pgno: 323) [L2]  
a) to ensure that at least one of the necessary conditions cannot hold  
b) to ensure that all of the necessary conditions do not hold  
c) to decide if the requested resources for a process have to be given or not  
d) to recover from a deadlock***

***Answer: a***

***49. If deadlocks occur frequently, the detection algorithm must be invoked \_\_\_\_\_\_\_\_(Pgno: 336) [L1]  
a) rarely  
b) frequently  
c) rarely & frequently  
d) casually***

***Answer: b***

***50. A system has 3 processes sharing 4 resources. If each process needs a maximum of 2 units then, deadlock \_\_\_\_\_\_\_\_\_\_\_\_ (PgNo:CH-7) [L2]  
a) can never occur  
b) may occur  
c) has to occur  
d) cannot tell***

***Answer: a***

***51. Which among the following is helpful in CPU scheduling decisions?***

***a. When a process switches from thewaiting state to the running state***

***b. When a process switches from the ready state to the running state***

***c.When a process switches from the waiting state to the ready state***

***d.When a process begins (Pgno: 203) [L2]***

***Answer: c***

***PART B***

***4 MARKS:***

1. ***What is critical section problem? (PgNo:263) [L1]***
   * ***Consider a system consists of ‘n‘processes. Each process has segment of Code called a critical section, in which the process may be changing common variables, updating a table, writing a file.***
   * ***When one process is executing in its critical section, no other process can allowed to execute in its critical section.***
2. ***Define busy waiting and spinlock. (Pgno:213) [L1]***
   * ***When a process is in its critical section, any other process that tries to enter its critical section must loop continuously in the entry code.***
   * ***This is called as busy waiting and this type of semaphore is also called a spinlock, because the process while waiting for the lock.***
3. ***What are the requirements that a solution to the critical section problem must satisfy? (Pgno:207) [L1]***

***The three requirements are,***

* + ***Mutual Exclusion***
  + ***Progress***
  + ***Bounded Problem***

1. ***Define entry section and exit section. (Pgno:207) [L1]***
   * ***The critical section problem is to design a protocol that the processes can use to cooperate.***
   * ***Each process must request permission to enter its critical section. The section of the code implementing this request is the entry section.***
   * ***The critical section is followed by an exit section. The remaining code is the remainder section.***
2. ***Define semaphores. (Pgno:214) [L1]***
   * ***Semaphore is a synchronization toll. A semaphore S is an integer variable that apart from initialization is accessed only through 2 standard atomic operations.***
     1. ***Wait***
     2. ***Signal***
3. ***Name dome classic problem of synchronization? (Pgno:219) [L1]***
   * ***The Bounded – Buffer Problem***
   * ***The Reader – Writer Problem***
   * ***The Dining –Philosophers Problem***
4. ***What is the concept behind strong semaphore and spinlock?(Pgno:214) [L2]***
   * ***A semaphore is a generalization of a lock (or, the other way around, a lock is a special case of a semaphore).***
   * ***Spinlocks are only valid within one process whereas semaphores can be used to synchronize between different processes, too***
5. ***Elucidate mutex locks with its procedure.(Pgno:206) [L2]***
   * ***A mutual exclusion object (mutex) is a program object that allows multiple program threads to share the same resource, such as file access, but not simultaneously.***
   * ***After this stage, any thread that needs the resource must lock the mutex from other threads while it is using the resource.***
6. ***Can a mutex be locked more than once ? (Pgno:224) [L2]***
   * ***a recursive mutex can be locked more than once (POSIX complaint systems), in which a count is associated with it, yet retains only one state (locked/unlocked).***
   * ***The programmer must unlock the mutex as many number times as it was locked.***
7. ***What are the different synchronization mechanisms? (Pgno:209) [L1]***
   * ***There are two types of synchronization: data synchronization and process synchronization:***
   * ***The simultaneous execution of multiple threads or processes to reach a handshake such that they commit a certain sequence of actions.***
   * ***Lock, mutex, and semaphores are examples of process synchronization.***
8. ***Define classical problem of synchronization(Pgno:207) [L1]***
   * ***Bounded buffer problem or producer-consumer problem is a classical synchronization problem***
   * ***we have a buffer with n cells or n slots and there are 2 process producers and consumers can produce and consume one article at a time.***
9. ***Why is process synchronization required? (Pgno:210) [L2]***
   * ***The need for synchronization originates when processes need to execute concurrently.***
   * ***The main purpose of synchronization is the sharing of resources without interference using mutual exclusion.***
   * ***The other purpose is the coordination of the process interactions in an operating system.***
10. ***List out the circumstances under which CPU scheduling decisions takes place. (PgNo:263) [L1]***

* ***When a process switches from the running state to the waiting state (for example, as the result of an I/O request or an invocation of wait()for the termination of a child process)***
* ***When a process switches from the running state to the ready state (for example, when an interrupt occurs)***
* ***When a process switches from the waiting state to the ready state (for example, at completion of I/O)***
* ***When a process terminates***

1. ***Differentiate preemptive and non-preemptive scheduling. (PgNo:264) [L2]***

* ***In non-preemptive scheduling, once the CPU has been allocated to a process, the process keeps the CPU until it releases the CPU either by terminating or by switching to the waiting state.***
* ***Otherwise it is preemptive scheduling.***

1. ***Define dispatcher and state its functions. (PgNo:265) [L1]***

* ***The dispatcher is the module that gives control of the CPU to the process selected by the short-term scheduler.***
* ***This function involves the following:***
  + ***Switching context***
  + ***Switching to user mode***
  + ***Jumping to the proper location in the user program to restart that program***

1. ***Suggest the criteria’s for comparing CPU scheduling algorithms. (PgNo:265) [L1]***

* ***CPU utilization.***
* ***Throughput***
* ***Turnaround time***
* ***Waiting time***
* ***Response time.***

1. ***Is there any advantage in having different time – quantum sizes on different levels of a multi-level queuing system? Justify. (PgNo:275) [L2]***

* ***The idea is to separate processes according to the characteristics of their CPU bursts.***
* ***Processes that need more frequent servicing, for instance, interactive processes such as editors, can be in a queue with a small time quantum.***
* ***Processes with no need for frequent servicing can be in a queue with a larger quantum, requiring fewer context switches to complete the processing, and thus making more efficient use of the computer.***

1. ***What are the parameters present in multilevel feedback queue scheduler? (PgNo:276) [L1]***

* ***The number of queues***
* ***The scheduling algorithm for each queue***
* ***The method used to determine when to upgrade a process to a higher-priority queue***
* ***The method used to determine when to demote a process to a lower-priority queue***
* ***The method used to determine which queue a process will enter when thatprocess needs service***

1. ***Define Deadlock. Specify the sequence in which process utilize a resource.( (PgNo:315,316)[L1]***

* ***waiting process is never again able to change state, because the resources it has requested are held by other waiting processes. This situation is called a deadlock.***
* ***Sequence***
  + ***Request***
  + ***Use.***
  + ***Release.***

1. ***List the necessary conditions for deadlock situation to occur. (PgNo:319) [L1]***

* ***Mutual exclusion.***
* ***Hold and wait***
* ***No preemption***
* ***Circular wait.***

1. ***When a state is said to be safe? (PgNo:328) [L2]***

* ***A state is safe if the system can allocate resources to each process (up to its maximum) in some order and still avoid a deadlock.***
* ***System is in a safe state only if there exists a safe sequence.***

1. ***List three examples of deadlock in day today life. (PgNo:325) [L2]***

***(Answer from website)***

* ***Two cars crossing a single lane bridge from opposite directions.***
* ***A person going down a ladder while another person is climbing up the ladder.***
* ***Two trains traveling toward each other on the same track.***

1. ***Is it possible to have deadlock involving only one process? Justify.( PgNo:341)***

* ***It is not possible to have a deadlock involving only one single process.***
* ***The deadlock involves a circular “hold-and-wait” condition between two or more processes.***
* ***“one” process cannot hold a resource yet be waiting for another resource that it is holding. So for one process, deadlock cannot be possible***

1. ***State the issues need to be addressed if preemption is required to deal with deadlocks. (PgNo:339) [L1]***

* ***selecting a victim,***
* ***rollback,***
* ***starvation.***
* ***In a system that selects victims for rollback primarily on the basis of cost factors, starvation may occur, and the selected process can never complete its designated task***

1. ***Why deadlock avoidance is necessary? (PgNo:339) [L2]***

* ***A method for avoiding deadlocks, rather han preventing them, requires that the operating system have a priori information about how each process will utilize system resources.***
* ***The banker ’s algorithm( for example)***

**PART C**

**12 MARKS:**

1. ***Outline a solution using semaphores to solve dinning philosopher problem.(Pgno:215) [L3]***
2. ***Show how wait () and signal() semaphore operations could be implemented in multiprocessor environments, using Test and Set instructions. The solution should exhibit minimal busy waiting. Develop pseudo code for implementing operations. (Pgno:226) [L2]***
3. ***Briefly discuss about Peterson’s solution.(Pgno:207)[L1]***
4. ***(i)Explain the dining philosopher’s critical section problem solution using monitor. (8) (Pgno:222)[L1]***

***(ii)Write the algorithm using test-and –set () instruction that satisfy all the critical section requirements. (4)(Pgno:222)[L2]***

1. ***What is the important feature of critical section? State the dining philosopher’s problem and show how to allocate the several resources among several processes in a deadlock and starvation free manner.(12) (pgno:221)[L3]***
2. ***With an example elaborate Classical Problems of synchronization.(12) (pgno:219)[L1]***
3. ***Compute non-preemptive SJF scheduling algorithm (PgNo:267) [L2]***

| ***Process*** | ***Arrival time*** | ***Burst time*** |
| --- | --- | --- |
| ***P1*** | ***0*** | ***7*** |
| ***P2*** | ***2*** | ***4*** |
| ***P3*** | ***4*** | ***1*** |
| ***P4*** | ***5*** | ***4*** |
| ***P5*** | ***3*** |  |

1. ***Consider the following set of processes with the length of the CPU-burst time in given***

***ms:***

| ***Process*** | ***Arrival time*** | ***Burst time*** |
| --- | --- | --- |
| ***P1*** | ***8*** | ***0*** |
| ***P2*** | ***4*** | ***1*** |
| ***P3*** | ***9*** | ***2*** |
| ***P4*** | ***5*** | ***3*** |
| ***P5*** | ***3*** | ***4*** |

***Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF, priority and RR (quantum=2) scheduling. Also calculate waiting time and turnaround time for each scheduling algorithms. (PgNo: 266) [L2]***

1. ***Discuss how the following pairs of scheduling criteria conflict in certain settings***

***i) CPU utilization and response time***

***ii) Average turnaround time and maximum waiting time.***

***iii)I/O device utilization and CPU utilization(PgNo:265) [L1]***

1. ***Explain Deadlock detection with suitable example.(PgNo:333) [L2]***
2. ***Illustrate Bankers algorithm with an example (PgNo:330) [L2]***
3. ***Describe Deadlock prevention in detail. (PgNo:323) [L2]***
4. ***Explain the methods for handling deadlocks. (PgNo:322) [L1]***

***\*\*\*\*\*\*\*\*\*\****