**2019**

Q.41 Consider the following four processes with arrival times (in milliseconds) and their length of CPU bursts (in milliseconds) as shown below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process | P1 | P2 | P3 | P4 |
| Arrival time | 0 | 1 | 3 | 4 |
| CPU burst time | 3 | 1 | 3 | Z |

These processes are run on a single processor using preemptive Shortest Remaining Time First scheduling algorithm. If the average waiting time of the processes is 1 millisecond, then the value of Z is .

|  |  |
| --- | --- |
| Q.42 | The index node (inode) of a Unix-like file system has 12 direct, one single-indirect and one double-indirect pointers. The disk block size is 4 kB, and the disk block address is 32-bits  long. The maximum possible file size is (rounded off to 1 decimal place) \_ GB. |

**2018**

1. Consider a process executing on an operating system that uses demand paging. The average time for a memory access in the system is *M* units if the corresponding memory page is available in memory, and *D* units if the memory access causes a page fault. It has been experimentally measured that the average time taken for a memory access in the process is *X* units.

Which one of the following is the correct expression for the page fault rate experienced by the process?

(A) (*D* – *M*) / (*X* – *M*) (B) (*X* – *M*) / (*D* – *M*)

(C) (*D* – *X*) / (*D* – *M*) (D) (*X* – *M*) / (*D* – *X*)

* 1. Consider a system with 3 processes that share 4 instances of the same resource type. Each process can request a maximum of 𝐾 instances. Resource instances can be requested and released only one at a time. The largest value of 𝐾 that will always avoid deadlock is .
  2. In a system, there are three types of resources: *E, F* and *G*. Four processes *P0, P1, P2* and *P3* execute concurrently. At the outset, the processes have declared their maximum resource requirements using a matrix named Max as given below. For example, Max[*P2,F*] is the maximum number of instances of *F* that *P2* would require. The number of instances of the resources allocated to the various processes at any given state is given by a matrix named Allocation.

Consider a state of the system with the Allocation matrix as shown below, and in which 3 instances of *E* and 3 instances of *F* are the only resources available.

|  |  |  |  |
| --- | --- | --- | --- |
| Allocation | | | |
|  | *E* | *F* | *G* |
| *P0* | 1 | 0 | 1 |
| *P1* | 1 | 1 | 2 |
| *P2* | 1 | 0 | 3 |
| *P3* | 2 | 0 | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Max | | | |
|  | *E* | *F* | *G* |
| *P0* | 4 | 3 | 1 |
| *P1* | 2 | 1 | 4 |
| *P2* | 1 | 3 | 3 |
| *P3* | 5 | 4 | 1 |

From the perspective of deadlock avoidance, which one of the following is true?

1. The system is in *safe* state.
2. The system is not in *safe* state, but would be *safe* if one more instance of *E* were available
3. The system is not in *safe* state, but would be *safe* if one more instance of *F* were available
4. The system is not in *safe* state, but would be *safe* if one more instance of *G* were available
   1. Consider the following solution to the producer-consumer synchronization problem. The shared buffer size is 𝑁. Three semaphores *empty*, *full* and mutex are defined with respective initial values of 0, 𝑁 and 1. Semaphore *empty* denotes the number of available slots in the buffer, for the consumer to read from. Semaphore *full* denotes the number of available slots in the buffer, for the producer to write to. The placeholder variables, denoted by P, Q, R, and S, in the code below can be assigned either *empty* or *full*. The valid semaphore operations are: wait() and signal()*.*

|  |  |
| --- | --- |
| Producer: | Consumer: |
| do{  wait(P); wait(mutex);  //Add item to buffer signal(mutex); signal(Q);  }while(1); | do{  wait(R); wait(mutex);  //Consume item from buffer signal(mutex);  signal(S);  }while(1); |

Which one of the following assignments to P, Q, R and S will yield the correct solution?

1. P: *full*, Q: *full*, R: *empty*, S: *empty*
2. P: *empty*, Q: *empty*, R: *full*, S: *full*
3. P: *full*, Q: *empty*, R: *empty*, S: *full*
4. P: *empty*, Q: *full*, R: *full*, S: *empty*
   1. Consider a storage disk with 4 platters (numbered as 0, 1, 2 and 3), 200 cylinders (numbered as 0, 1, … , 199), and 256 sectors per track (numbered as 0, 1, … , 255). The following 6 disk requests of the form [sector number, cylinder number, platter number] are received by the disk controller at the same time:

[120, 72, 2] , [180, 134, 1] , [60, 20, 0] , [212, 86, 3] , [56, 116, 2] , [118, 16, 1]

Currently the head is positioned at sector number 100 of cylinder 80, and is moving towards higher cylinder numbers. The average power dissipation in moving the head over 100 cylinders is 20 milliwatts and for reversing the direction of the head movement once is 15 milliwatts. Power dissipation associated with rotational latency and switching of head between different platters is negligible.

The total power consumption in milliwatts to satisfy all of the above disk requests using the Shortest Seek Time First disk scheduling algorithm is .