MADE EASY
India's Best Institute for IES, GATE & PSUs

Ashima Garg
(Enroll User)

Sign out

Ashima Garg
Course: GATE
Computer Science Engineering(CS)

- HOME
- MY TEST
- BOOKMARKS
- MY PROFILE
- REPORTS
- BUY PACKAGE
- ASK AN EXPERT
- OFFER
- EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

OPERATING SYSTEM-2 GATE 2019 - REPORTS

OVERALL ANALYSIS COMPARISON REPORT **SOLUTION REPORT**

ALL(17) CORRECT(14) INCORRECT(3) SKIPPED(0)

Q. 1

A CPU scheduling algorithm find the order of execution of its scheduled processes. Assume '*n*' processes are scheduled on one processor, then what is different number of schedules that are possible?

Solution Video | Have any Doubt ?

A
n

B
$n^2 - n$

C
n^2

D
$n!$

Your answer is **Correct**

Solution :
(d)
Since only single processor is given, then any process can be scheduled any time and at any instance only one process can be scheduled i.e. n! ways are possible for scheduling.

QUESTION ANALYTICS

Q. 2

Which of the following operations require the executing code to be operating with Kernel mode?

Solution Video | Have any Doubt ?

A
Reading status of process

B
Reading time of clock

C
Disabling interrupt

Correct Option

Solution :
(c)
Disabling interrupt is done in Kernel mode rest can be executed in user mode.
So, option (c) is answer.

D
Both (b) and (c)

Your answer is **Wrong**

QUESTION ANALYTICS

Ashima Garg

Course: GATE

Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

wait decrements the semaphore value, and signal increments the semaphore value. The pseudocode of *A* and *B* is as follows:

```
A :                          B :
while true do                while true do
{                            {
    $S_1$;                       $S_3$;
    $S_2$;                       $S_4$;
    $X = X + 1$;                 $Y = Y + 1$;
    $Y = Y - 1$;                 $X = X - 1$;
    Signal ($P$);                Signal ($Q$);
    Signal ($Q$);                Signal ($P$);
}                            }
```

In order to satisfy mutual exclusion, bounded waiting and progress condition the correct operators at S_1, S_2, S_3 and S_4 are

FAQ | Solution Video | Have any Doubt ?

A

Wait (*P*) Wait (*Q*) Wait (*Q*) Wait (*P*)

B

Wait (*P*) Wait (*Q*) Wait (*P*) Wait (*Q*)

Correct Option

Solution :

(b)

Option (a) and (c) do not satisfy progress condition since (a), (c) leads to deadlock.
Option (b) is correct to satisfy progress condition.

C

Wait (*Q*) Wait (*P*) Wait (*P*) Wait (*Q*)

Your answer is **Wrong**

D

None of these

QUESTION ANALYTICS

Q. 4

What is the main reason for occurrence of RACE condition while synchronizing the processes?

FAQ | Solution Video | Have any Doubt ?

A

The two processes are trying to update the variable at the same time.

B

More than one process entering into critical section at the same time.

C

Mutual exclusion condition not satisfies.

D

All of these

Your answer is **Correct**

Solution :

(d)

Meaning of all statements (a), (b), (c) is logically same. So answer is (d).

QUESTION ANALYTICS

Q. 5

Ashima Garg

Course: GATE

Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

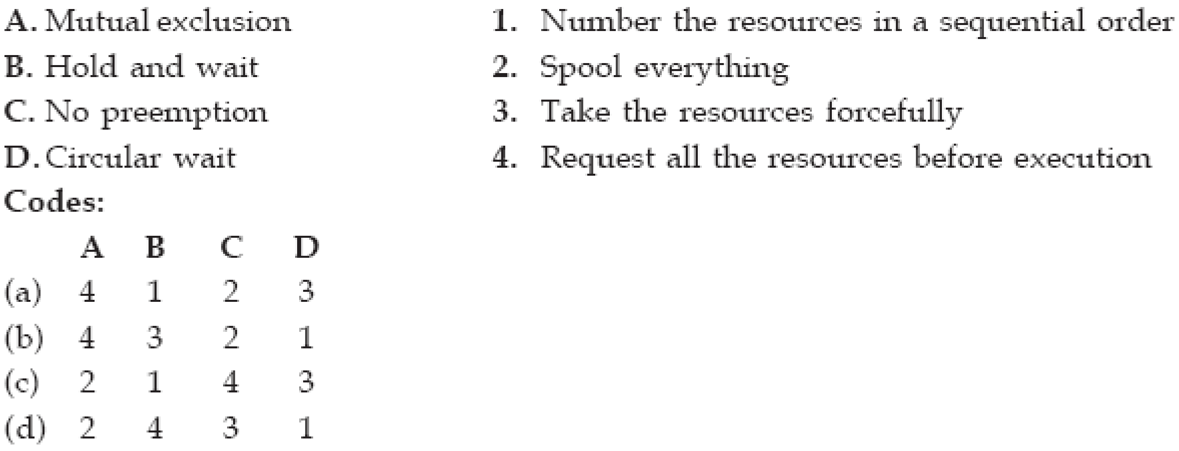
REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES



A. Mutual exclusion
B. Hold and wait
C. No preemption
D. Circular wait

1. Number the resources in a sequential order
2. Spool everything
3. Take the resources forcefully
4. Request all the resources before execution

Codes:

	A	B	C	D
(a)	4	1	2	3
(b)	4	3	2	1
(c)	2	1	4	3
(d)	2	4	3	1

FAQ | Solution Video | Have any Doubt ?

A
a

B
b

C
c

D
d

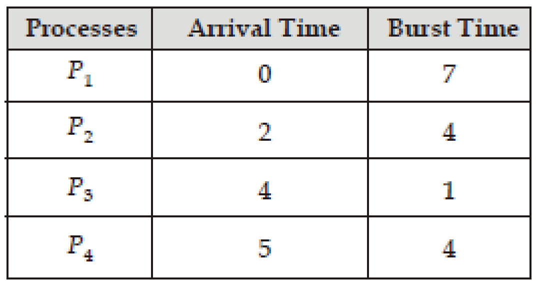
Your answer is **Correct**

Solution :
(d)
Considering each option of List-I:
• Mutual exclusion can be solved by spooling everything.
• Hold and wait can be solved by requesting all the resources before hand.
• No preemption can be solved by taking request away or by releasing all resources.
• Circular wait can be solved by numbering the resources in some order.

QUESTION ANALYTICS

Q. 6

An operating system uses Shortest Remaining Times First (SRTF) scheduling algorithm for preemptive scheduling of processes. Consider the following set of processes with their arrival times and the CPU burst times in msec.



Processes	Arrival Time	Burst Time
P_1	0	7
P_2	2	4
P_3	4	1
P_4	5	4

The average waiting time (msec) of the processes is _______.

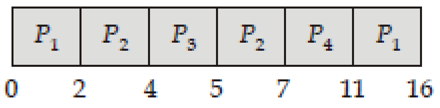
Solution Video | Have any Doubt ?

3

Your answer is **Correct**3

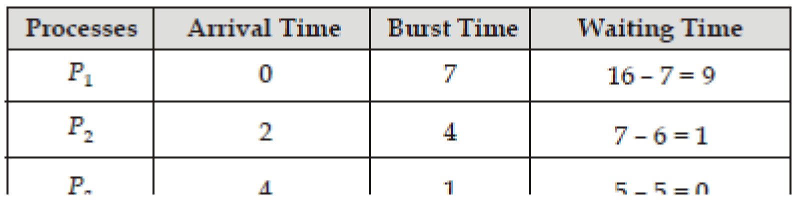
Solution :
3

Gantt chart:



P_1	P_2	P_3	P_2	P_4	P_1

0 2 4 5 7 11 16



Processes	Arrival Time	Burst Time	Waiting Time
P_1	0	7	16 – 7 = 9
P_2	2	4	7 – 6 = 1
P_3	4	1	5 – 5 = 0

MADE EASY India's Best Institute for IES, GATE & PSUs

Ashima Garg
(Enroll User)

Sign out

Ashima Garg
Course: GATE
Computer Science Engineering(CS)

- HOME
- MY TEST
- BOOKMARKS
- MY PROFILE
- REPORTS
- BUY PACKAGE
- ASK AN EXPERT
- OFFER
- EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

QUESTION ANALYTICS

Q. 7

At a particular time of computation the value of a counting semaphore is ‘x’. Then 22 V operations 12 P operations and 3 V operations were completed on this semaphore, if the resulting value of semaphore is 20 then the value of x is _______.

Solution Video | Have any Doubt ?

7

Your answer is **Correct**7

Solution :
7

Initial value of semaphore = x

3^{rd} step: 22 V operations = $x + 22$

2^{nd} step: 12 P operations = $x + 22 - 12 = x + 10$

1^{st} step: 3 V operations = $x + 10 + 3 = x + 13$

As per question, $x + 13 = 20$

$\Rightarrow \quad x = 7$

QUESTION ANALYTICS

Q. 8

A process spends 30% of its execution time waiting for completion of I/O operation. If there are 5 processes in memory at once, then the probability of CPU time utilized is ______ %. (Assume all I/O operations are overlapped). (Upto 2 decimal places)

FAQ | Have any Doubt ?

99.75 (98.75 - 100.75)

Your answer is **Correct**99

Solution :
99.75 (98.75 - 100.75)

When there are ‘n’ processes competing with waiting time ‘t’.

Then probability of CPU time utilization = $1 - t^n$.

CPU time utilization $= (1 - 0.3^5) = 0.9975 = 99.75\%$

QUESTION ANALYTICS

Q. 9

A system has 13 identical resources and ‘n’ process competing for them. Each process can request atmost 5 resources. The minimum possible value of ‘n’ which lead to deadlock is _______.

Solution Video | Have any Doubt ?

4

Your answer is **Correct**4

Solution :
4

$$\begin{aligned} \text{Resources} &= 13 \\ \text{Process} &= n \\ \text{Requirement} &= \text{Max} = 5 \\ [n-1] \times 4 + 5 &= 13 \quad \text{[Ensuring atleast one process has 5 resources, i.e. no dead} \\ 4n - 4 + 5 &= 13 \end{aligned}$$

Ashima Garg
Course: GATE
Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

QUESTION ANALYTICS

Q. 10

Consider four process all are arriving at time zero, with total execution time of 20, 10, 10 and 20 unit respectively. Each process spends the first 20% of execution time doing CPU, the next 60%of doing $\mathrm{I/O}$ computation and the last 20% of time doing CPU computation again. The operating system uses longest time remaining first scheduling algorithm (LRTF) and schedules a new process either when running process get blocked $\mathrm{I/O}$ or when the running process finishes its CPU burst. Assume that are I/O operations can be overlapped as much as possible. What is the completion time of process P_0? [Note: When same burst occurs for multiple process high priority given to lowest process id].

FAQ Solution Video Have any Doubt ?

A
18

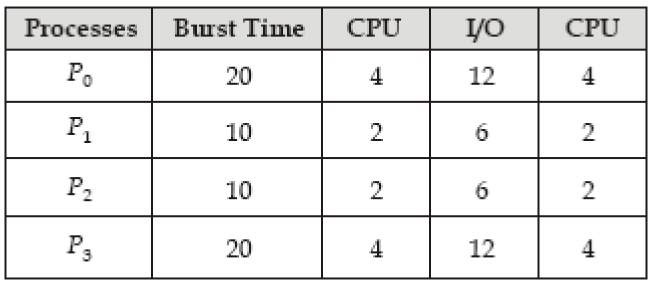
B
20

C
24

D
28

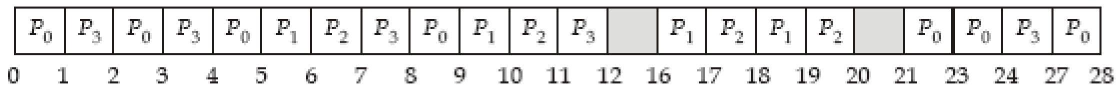
Your answer is **Correct**

Solution :
(d)



Processes	Burst Time	CPU	I/O	CPU
P_0	20	4	12	4
P_1	10	2	6	2
P_2	10	2	6	2
P_3	20	4	12	4

Gantt Chart:



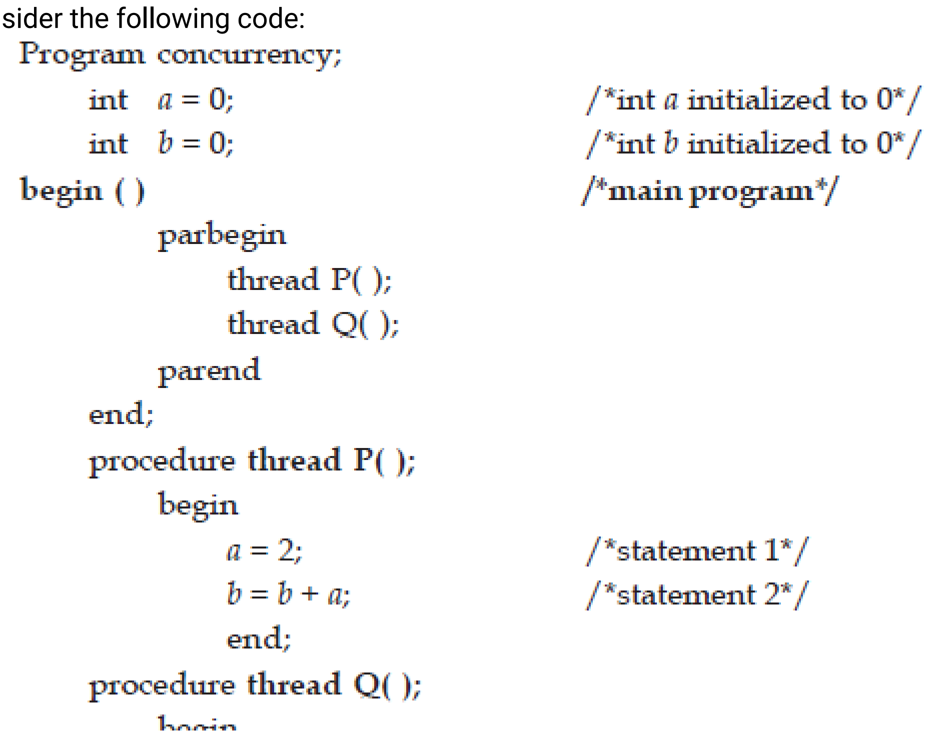
P_0	P_3	P_0	P_3	P_0	P_1	P_2	P_3	P_0	P_1	P_2	P_3		P_1	P_2	P_1	P_2		P_0	P_0	P_3	P_0

0 1 2 3 4 5 6 7 8 9 10 11 12 16 17 18 19 20 21 23 24 27 28

QUESTION ANALYTICS

Q. 11

Consider the following code:



```
    Program concurrency;
        int  a = 0;                 /*int a initialized to 0*/
        int  b = 0;                 /*int b initialized to 0*/
    begin ( )                       /*main program*/
            parbegin
                thread P( );
                thread Q( );
            parend
        end;
        procedure thread P( );
            begin
                a = 2;              /*statement 1*/
                b = b + a;          /*statement 2*/
                end;
        procedure thread Q( );
            begin
```

MADE EASY
India's Best Institute for IES, GATE & PSUs

Ashima Garg
(Enroll User)

Sign out

Ashima Garg
Course: GATE
Computer Science Engineering(CS)

- HOME
- MY TEST
- BOOKMARKS
- MY PROFILE
- REPORTS
- BUY PACKAGE
- ASK AN EXPERT
- OFFER
- EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

Suppose a process has 2 concurrent threads; one thread executes statement 1 and 2 and other thread executes statement 3 and 4. What are the possible values of variable '*a*' and '*b*' when the code finishes execution?

FAQ | Solution Video | Have any Doubt ?

A

$a = \{2, 4, 8\}$
$b = \{4, 6, 12, 14\}$

B

$a = \{2, 6, 8\}$
$b = \{4, 6, 12\}$

C

$a = \{2, 8\}$
$b = \{4, 6, 12\}$

Your answer is **Correct**

Solution :
(c)
For variable 'a' :

$$\begin{aligned}
\text{I.} \qquad & \text{(i) } a = 2 \\
& \text{(ii) } a = 2 + 6 = 8 \\
& \text{Sequence} = \{1, 2, 3, 4\} \text{ or } \{1, 3, 2, 4\} \\
\text{II.} \qquad & \text{(iii) } a = 0 + 6 = 6 \\
& \text{(iv) } a = 2 \\
& \text{Sequence} = \{3, 4, 2, 1\}
\end{aligned}$$

Hence possible value are $\{2, 8\}$.
For variable 'b' :

$$\begin{aligned}
\text{I.} \qquad & \text{(i) } b = 0 + 2 = 2 \\
& \text{(ii) } b = 4 \\
& \text{Sequence} = \{1, 2, 3, 4\} \\
\text{II.} \qquad & \text{(i) } b = 4 \\
& \text{(ii) } b = 4 + 2 = 6 \\
& \text{Sequence} = \{3, 1, 2, 4\} \text{ or } \{3, 4, 1, 2\} \text{ or } \{1, 3, 2, 4\} \\
\text{III.} \qquad & \text{(i) } b = 4 \\
& \text{(ii) } b = 4 + 8 = 12 \\
& \text{Sequence} = \{3, 1, 4, 2\}
\end{aligned}$$

Hence possible value are $\{4, 6, 12\}$.

D

$a = \{2, 8\}$
$b = \{4, 12\}$

QUESTION ANALYTICS

Q. 12

Consider the following problems:

P_1 : Starvation $\qquad$ P_2 : Deadlock
P_3 : Inconsistent data $\qquad$ P_4 : Priority inversion

Which of the above problem may occur due to the use of locking resources?

FAQ | Solution Video | Have any Doubt ?

A

P_1 , P_2 and P_3 only

B

P_1 and P_2 only

C

MADE EASY India's Best Institute for IES, GATE & PSUs

Ashima Garg
(Enroll User)

Sign out

Ashima Garg

Course: GATE

Computer Science Engineering(CS)

- HOME
- MY TEST
- BOOKMARKS
- MY PROFILE
- REPORTS
- BUY PACKAGE
- ASK AN EXPERT
- OFFER
- EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

All the given problems

Your answer is **Correct**

Solution :
(D)

Locking of resource can create:

1. **Starvation:** P_1 wants resource R_1, which is locked by process P_2, when process P_2 comp another process P_3 locks resource R_1 before process P_1, it will happen for indefinite tin process P_1 is in starvation.
2. **Inconsistent data:** This problem occur when one process fail in between and another deper process read updated value of failed process.
3. **Deadlock:** When two process both want two resources to complete but currently lock one resources. This will create deadlock since both are waiting for resource to be free.

QUESTION ANALYTICS

Q. 13

Consider two processes 'P_0' and 'P_1' need to access a critical section. Consider following synchronization construct used by both the processes:

```
Process '0'                          Process '1'
while (true) {                       while (true) {
flag[0] = true;                      flag[1] = true;
turn = 0;                            turn = 1;
while (flag[1] && (turn == 0));      while (flag[0] && (turn == 1));
    critical section                     critical section
flag[0] = false;                     flag[1] = false;
}                                    }
```

Here turn is a shared variable, which is initialized to '0'. Which of the following is true?

Solution Video | Have any Doubt ?

A

The proposed solution guarantees mutual exclusion but fail to prevent deadlock.

B

The proposed solution does not guarantees mutual exclusion but prevent deadlock.

C

The proposed solution guarantees mutual exclusion and prevent deadlock.

Your answer is **Correct**

Solution :
(c)
In given solution, atleast one process will enter into critical section i.e. process which arrive late enter first. So, not deadlock.
• Only one process can enter into critical section at any time because of checking of turn variable in while loop.

D

The proposed solution neither guarantees mutual exclusion nor prevent deadlock.

QUESTION ANALYTICS

Q. 14

Consider the following statements:

S_1 : Each thread can be scheduled separately.
S_2 : Switching between threads is faster.
S_3 : Blocking threads don't stop the entire process.
S_4 : Thread from same process can be run simultaneously on different processors.

Which of the above statement is an advantage of Kernal level threads over user level threads?

Solution Video | Have any Doubt ?

A

MADE EASY
India's Best Institute for IES, GATE & PSUs

Ashima Garg
(Enroll User)

Sign out

Ashima Garg
Course: GATE
Computer Science Engineering(CS)

- HOME
- MY TEST
- BOOKMARKS
- MY PROFILE
- REPORTS
- BUY PACKAGE
- ASK AN EXPERT
- OFFER
- EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

S_1 and S_3 only

Your answer is **Wrong**

C

S_1, S_3 and S_4 only

Correct Option

Solution :
(c)
Kernal level threads are designed as independent threads, so each thread can be scheduled separately.
• Kernal level threads have more context than user level threads, so switching between Kernal level threads is slower.
• Kernal level threads are designed as independent, so blocking one threads does not stop entire process, which is not the case with user level threads.
• Kernal level threads are independent so can be run simultaneously on different processors.

D

All the statements

QUESTION ANALYTICS

Q. 15

Consider the following processes, with the arrival time and the length of the CPU burst given in msec. The scheduling algorithm used is Round Robin with time quantum 15 msec.

Processes	Arrival Time	Burst Time
P_1	0	80
P_2	10	30
P_3	10	10
P_4	80	20
P_5	85	50

The average turn around time of these processes is _______ msec.

Solution Video | Have any Doubt ?

87

Your answer is **Correct**87

Solution :
87

Gantt chart:

P_1	P_2	P_3	P_1	P_2	P_1	P_4	P_5	P_1	P_4	P_5	P_1	P_5	P_1	P_5

0 15 30 40 55 70 85 100 115 130 135 150 165 180 185 190

Processes	Arrival Time	Burst Time	Turn Around Time
P_1	0	80	185
P_2	10	30	60
P_3	10	10	30
P_4	80	20	55
P_5	85	50	105
			Average = 87 msec

QUESTION ANALYTICS

Q. 16

Consider the following two concurrent processes with a shared memory variable x. Assume x initialized to 10.

MADE EASY India's Best Institute for IES, GATE & PSUs

Ashima Garg
(Enroll User)

Sign out

Ashima Garg

Course: GATE

Computer Science Engineering(CS)

- HOME
- MY TEST
- BOOKMARKS
- MY PROFILE
- REPORTS
- BUY PACKAGE
- ASK AN EXPERT
- OFFER
- EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

The maximum possible value of x when both processes have completed ________.

FAQ | Have any Doubt ?

255

Your answer is **Correct**255

Solution :
255

P_1 execute 10 interactions.

P_2 executes 9 iterations.

x has initial values is 10.

Therefore, final value is = [10 × 20 + 9 × 5] + 10 = 255

QUESTION ANALYTICS

Q. 17

A process execute the code:

```
main ( )
{
      fork ( );
      fork ( );
      fork ( );
      printf ("Hi");
}
```

The number of times "Hi" will be printed is _______.

FAQ | Solution Video | Have any Doubt ?

8

Your answer is **Correct**8

Solution :
8

Total 2^n – 1 child process created

$$2^3 - 1 = 7 + 1 \text{ parent}$$
$$= 8 \text{ times}$$

QUESTION ANALYTICS