Answer

RISHI KUMAR SONI ***1001774020***

Question 1 :

Thus Solution is ***incorrect.***

Reason :

Due to **interleaving of states**, switching between states i.e.

Producer and consumer takes place, So the value of the shared variable keeps on changing.

SCENARIO :

Steps :

1 . Producer accessing the buffer value and trying to change it's value.

2. Before changing the value ,context switch takes place.

3.Consumer access and update the buffer value.

4.Again Context switch occur.

5. Producer update the value now.

Thus the updated value in the buffer is wrong.

QUESTION 4 :

A)

S1 – 1

S2-3

S3-4

S4-2

S5-5

R1 – 2

R2-4

R3-5

R4-3

R5-6

B)

S1 – (1,0,0,0)

S2 – (1,0,2,0)

S3 – (1,0,3,0)

S4 – (2,0,0,0)

S5 – (1,0,2,2)

R1 – (1,0,1,0)

R2 – (1,0,2,1)

R3-(2,2,3,0)

R4-(2,1,0,0)

R5-(2,3,3,2)

C)

S5 concurrent with

S3, R4, R3, R5, S4

D)

Happen Before R3

S3, S1 , S4, R4, S2, R1

QUESTION 3:

In the given code :

This given Solution is ***incorrect.***

REASON :

There is no ***Enter Monitor*** in the deposit function.

And no ***Exit Monitor*** in the withdraw Function .

Counter Example :

```
Public void deposit ()
{
enter monitor(); // required
}
Public void withdraw ()
{
exit monitor();  // required
}
```

Thus Mutual exclusion cannot exits.

<u>QUESTION 2:</u>

Answer :

This is incorrect implementation.

Reason 1 :

The reason being , if we remove the synchronize method LINE 7 , then shared variable Permit will be access by the threads , any order .

As a result , ***FCFS will be FAILED .***

SCENARIO :

Before :

T1 COMES and AFTER T1 , T2 WILL ACCESS Line 11.

AFTER Removing Synchronized Method.

Lock is removed and as a result;

Any of the Thread can access the LINE 11.