CS & IT ENGINEERING



Operating System

Process Synchronization
DPP 05 (Discussion Notes)



By- Anjnee Bhatnagar ma'am



TOPICS TO BE COVERED

01 Question

02 Discussion



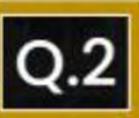
Choose the hardware type solution for the synchronization problem from the following



[MCQ]



- B. Strict alteration
- C. Peterson's solution
- D. Test and set instructions Hardware Solution



Consider the following two-process synchronization solution



X	Y
while (turn = = 1); $C.S$ $0 = = 1 \Rightarrow false$	while (turn = = 1); $C.S^{1}$ $O = = \Rightarrow false$
turn =1;	turn =1;

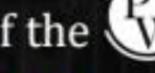
turn = 0

The shared variable turn is initialized to zero. Then the number of incorrect statements about above solution is/are? [NAT]

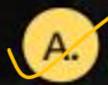
- (I) It violates mutual exclusion. True.
- (II) It violates progress. True
- (III) It violates bounded wait. True



The software type solution: strict alteration satisfies which of the



[MSQ]



Mutual exclusion



Progress X

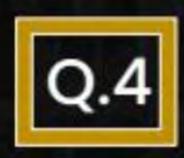
following?



Bounded wait



All of the above



Choose the correct statements regarding peterson's algorithm [MSQ]



Peterson's algorithm satisfies only mutual exclusion but not progress. $\Gamma_{\alpha|\beta}$



Peterson's algorithm is a solution for 2 processes.



Peterson's algorithm satisfies mutual exclusion, progress and bounded wait.

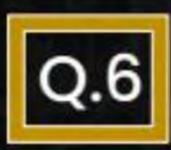


Peterson's algorithm is a solution for atleast 2 processes.



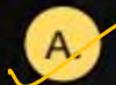
Which of the following is correct regarding solution to synchronization problem: [MCQ]

- A. Test and set is a 2-process, software based solution.
- B. Lock variable is a 2-process, hardware based solution.
- C. Strict alteration is a 2-process, hardware based solution.
- D. None of these.



Choose the characteristics for test and set synchronization solution.





It ensures mutual exclusion.



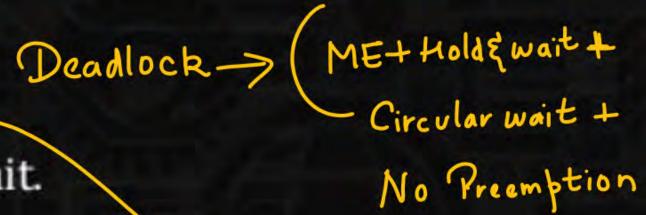
It is deadlock free.

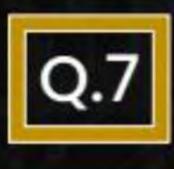


It does not guarantee bounded wait.



It may cause starvation.





For a implementing synchronization for atleast 2 processes (that should satisfy mutual exclusion, progress and bounded wait), which algorithm is best recommended?

[MCQ]

- A. Decker's algorithm
- B. Strick alteration
- C. Peterson's solution
- D. None of the above

2 or more



Which of the following is/are OS based synchronization mechanism?





Sleep() and wakeup()



Wait() and signal()



Monitor



Swap





