

NAME: KRUNAL RANIK

ROLL: U18C0081

CLASS: BTECH COMPS. 6TH SEM

Operating System Test 4

Ans 1: Thread is a small part of the process that executes some segment of the process.

A process can have multiple threads:

Some valid differences between a thread and a process are as follows:-

- Thread is more efficient in terms of context switching, terminating or creation than a process.
- Thread A process can have multiple threads but a thread belongs to only a single process.
- Processes are isolated whereas threads have shared memory.

Ans 2: A race condition occurs when outcome is depended on timing sequence of threads or processes.

Let's assume that Thread A executes withdraw. It reads the current amount as Rs 500. Let us assume thread B executes deposit which also assumes same current amount.

Now, if thread A withdraws ₹250 and writes ₹250 as remaining account and thread B deposits ₹250 and writes ₹250 as dep remaining amount which are both wrong. Hence, whichever transaction happened first writes a wrong value.

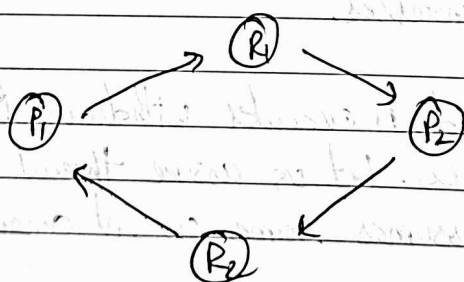
To prevent such conditions, we need synchronization such as mutex locks or semaphores.

Ans 3. A wait operation automatically decrements the value associated with the semaphore.

If two wait operations are executed on a semaphore, then when semaphore value is 1, the first process enters the critical zone while the other waits for the first process to call signal.

If they are not executed automatically, then it is possible that both operations might decrement semaphore value and hence, violate the principle of mutual exclusion.

Ans 4. Deadlock is a situation where one or more processes are waiting for each other to finish.
For example,



→ shows requirement of resource when process has process to resource otherwise having the resource.

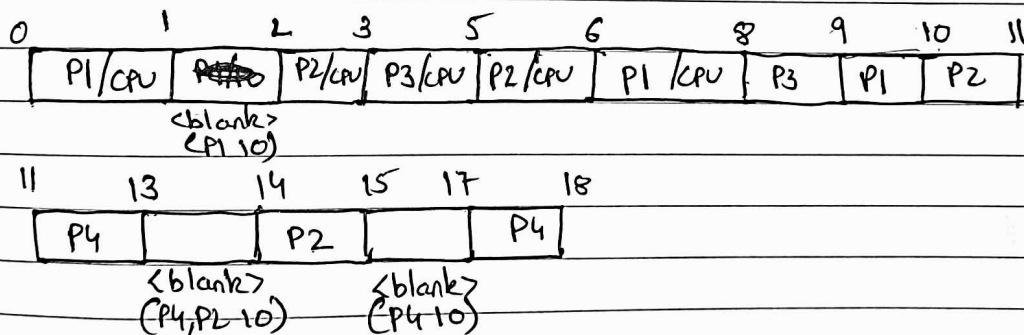
Some conditions of deadlock are as follows:-

- **Mutual Exclusion:-** The resource is held by only one process at a time.
In such a case some processes might be interdependent on each other.
- **Hold and Wait:-** A process can hold a number of resources at a time and at the same time can request for resources that are being held.

- No preemption:- A resource can't be preempted from the process by another process forcefully.
This leads to infinite waiting.
- Circular Wait:- Circular Wait is a condition when the first process is waiting for the resource held by the third process & so on.

Ans:-

a) The Gantt chart will be as follows:-



b) The ratio of CPU idle time is given by =
Total required time for CPU = $1+3+2+2+3+1+1+1$
= 14 units

Total time when processes completed = 18 units.

Idle time = $18 - 14 = 4$ units.

Idle time ratio = $\frac{4}{18} = \frac{2}{9} \approx 22\%$