

(RR) Round Robin

- used in OS, networks
- used to distribute resources among multiple users or processes
- each process or task given a time slice or quantum, which is a fixed amount of time during which the process can use the resource. The scheduler then switches to the next process in the queue after the time slice has elapsed regardless of whether the process has finished or not.
- This algorithm ensures that each process gets an equal share of resources and prevents any one process from monopolizing the resource.

round-Robin Scheduling is particularly useful in situations where there are multiple users or processes competing for the same resource and where fairness is a key

MULTI LEVEL QUEUE

→ Scheduling algorithm

• Used To organize MULTI levels of priority for Processes in computer system

- divide The Processes into different queues based on their Priority level and assigns different scheduling algorithm To each queue
- in a multi level queue scheduling algorithm, process are classified into different levels of priority based on resources requirement, response time, etc....)
- ensure That high Priority process executed first

Multilevel Feedback Queue

- Based on multilevel scheduling algorithm
- Processes are still classified according to priority but unlike multilevel queue scheduling, the queues are not fixed, instead the algorithm is dynamically adjusting the priority of each process based on its behavior and resource req.

موضوع الدرس اليوم التاريخ

PROCESS SYNCHRONIZATION

CONCEPT in OS That refers To The coordination of multiple processes or threads To ensure That They access shared resource in a mutually exclusive and consistent manner without IT may occur conflicts and data corruption

Critical section

problem,

→ a problem in concurrent programming that arises when multiple processes or threads access a shared resource or critical section in mutually ~~exclusive~~ exclusive manner.

- The critical section is a part of the code where a shared resource is accessed or modified.
- It's essential that only one process or thread executes the critical section at any given time to avoid conflicts.
- Can be described in three terms of requirements.

1 - Mutual Exclusion :- only one process or thread can access the critical section at a time

2 - Progress: if no process is in the critical section and one or more processes are waiting to enter, then only those processes that are not prevented to enter the critical section should be able to enter the critical section

3 . Bounded Waiting: There exists a limit on the no. of times other processes or threads are allowed to enter the critical section before the requesting process or thread is allowed to enter

