Round Robin Algorithm

PRESENTER NAME:

V. NAVEEN

B. RAMA KOTI REDDY

CH. AKHIL

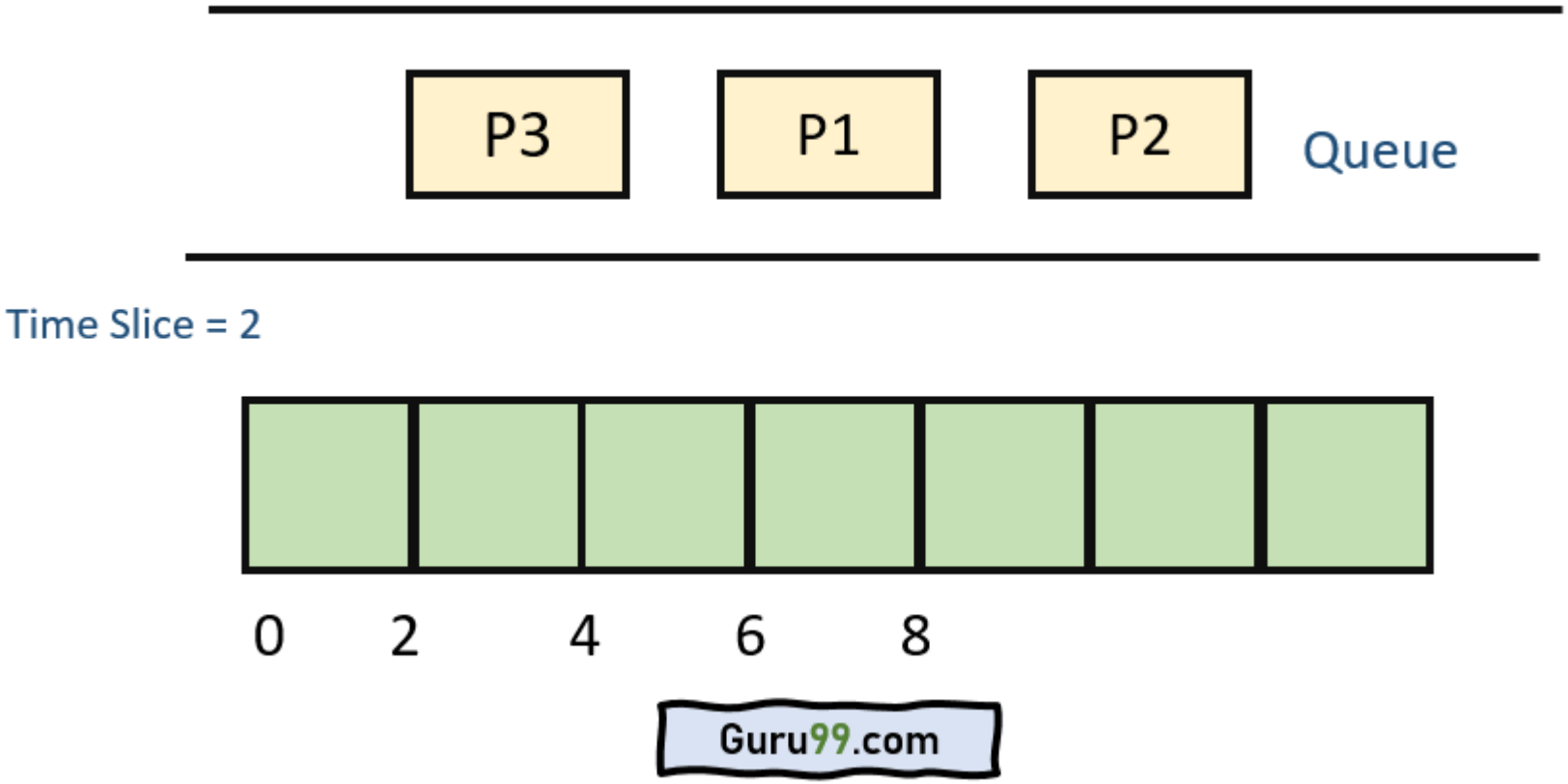
What is Round Robin Algorithm?

Round robin is a scheduling algorithm used in computing and data communication systems. It involves processing tasks or data sequentially, where each task or unit of data is assigned to a resource in a circular manner. This ensures that each resource gets an equal share of processing time or data access.

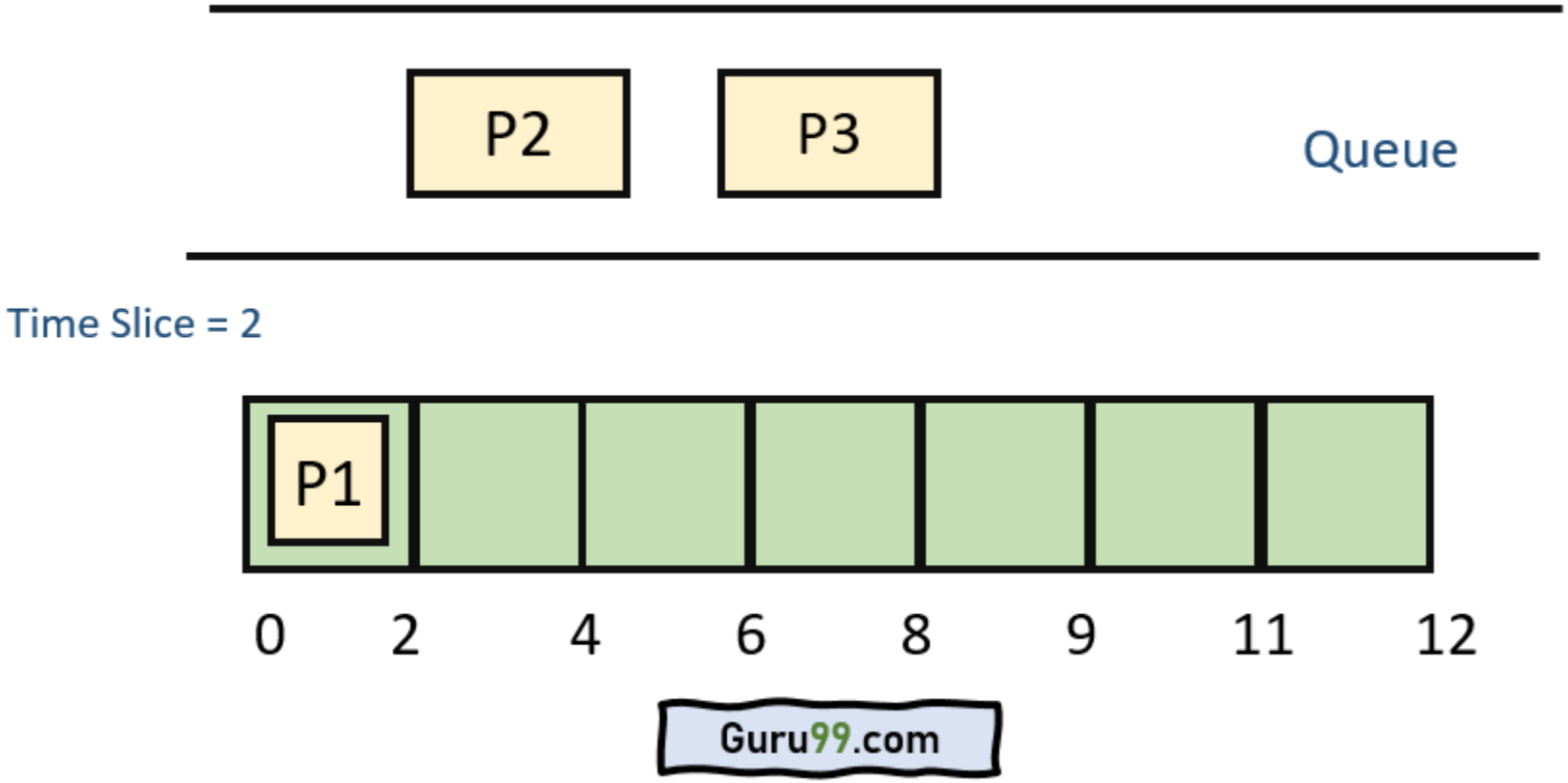
Characteristics of Round Robin scheduling

- A pre-emptive scheduling algorithm
- CPU shifts to next process after a fixed time interval known as time quantum or time-slice
- Pre-empted process are added to the end of the queue
- A hybrid and clock-driven model
- Time slice is usually the minimum but differs from OS to OS.
- Oldest , fairest, and easiest algorithm
- Widely used in traditional OS.

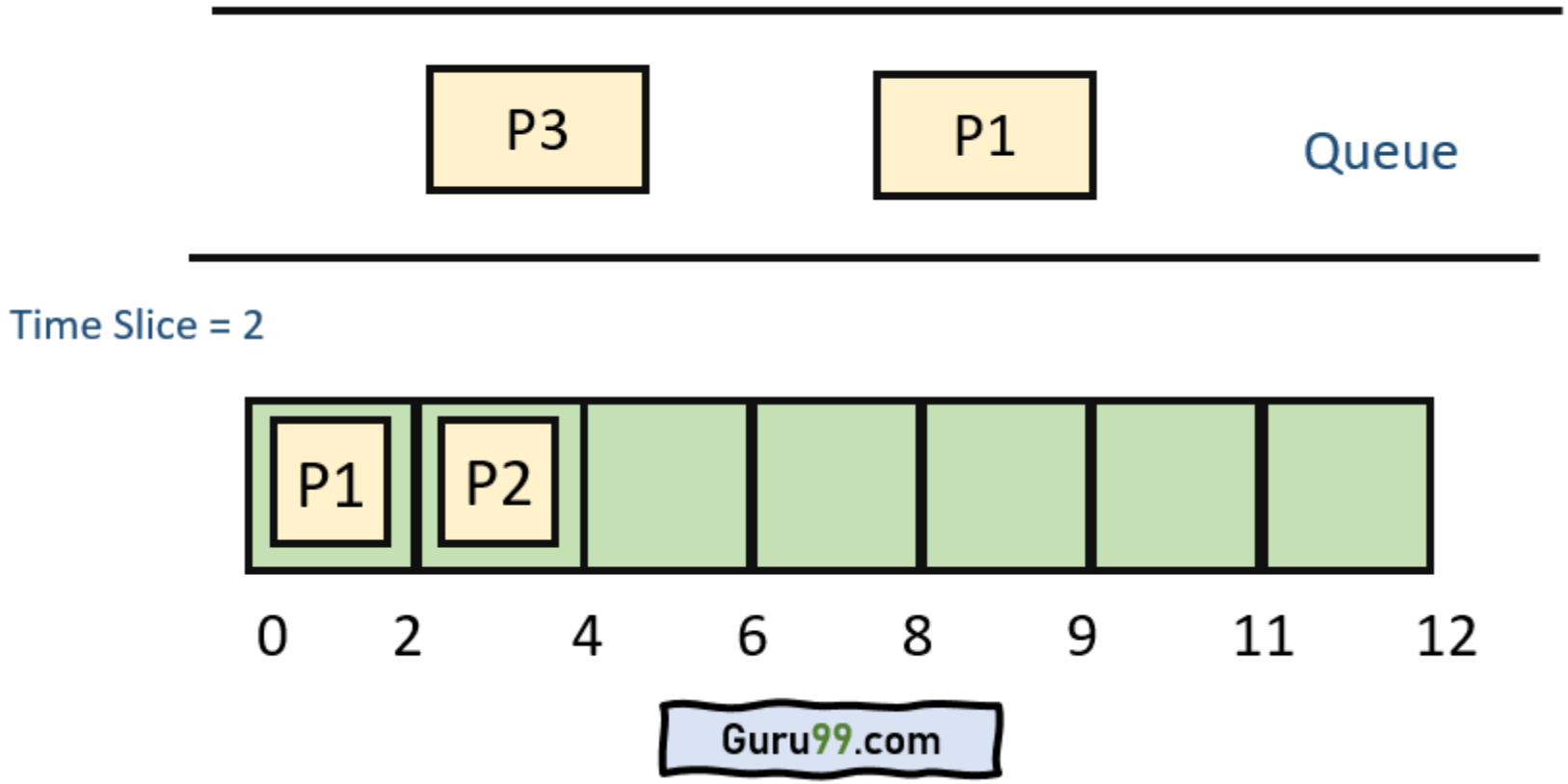
Examples of Round Robin Algorithm



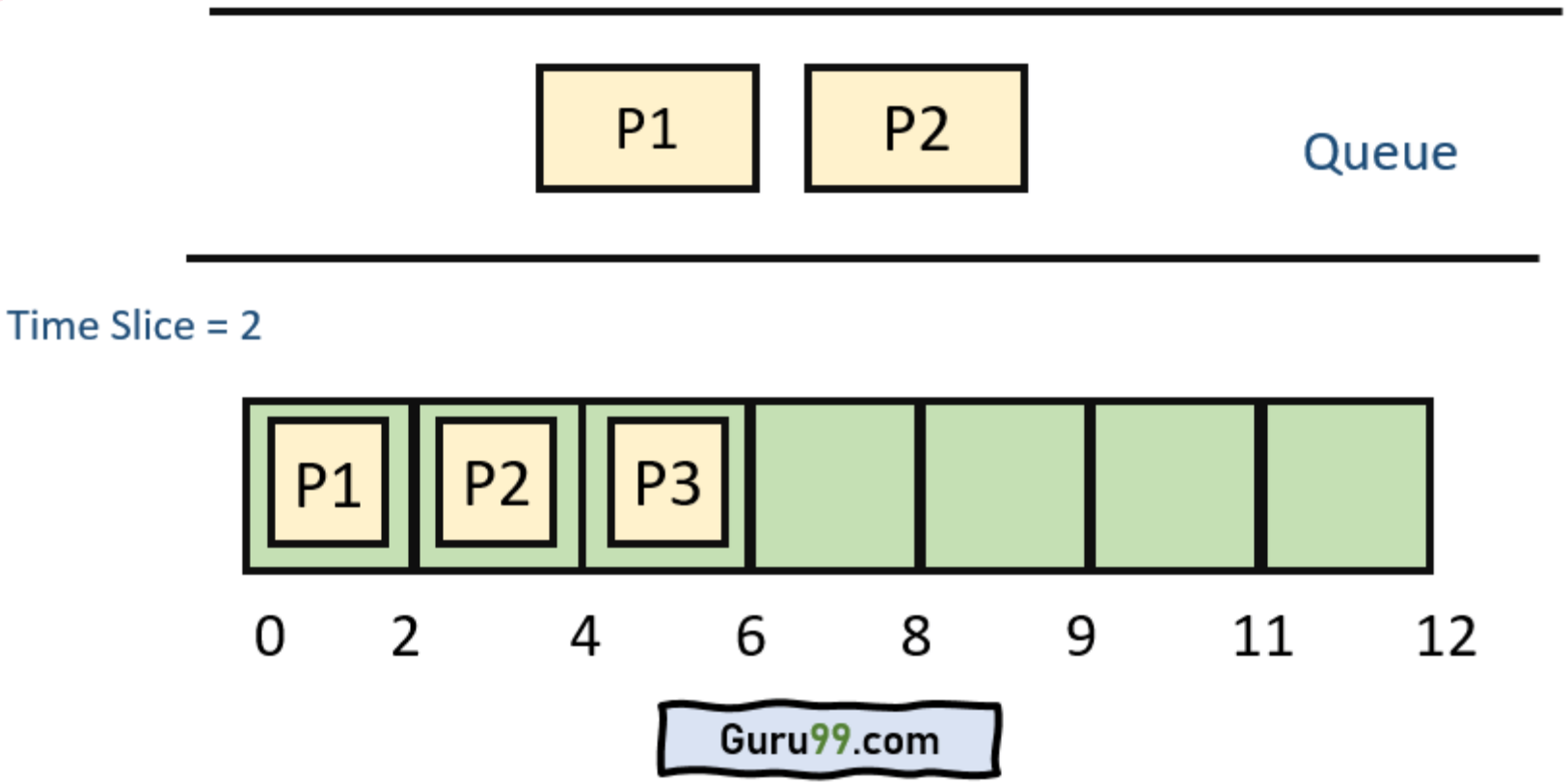
Step 1) The execution begins with process P1, which has burst time 4. Here, every process executes for 2 seconds. P2 and P3 are still in the waiting queue



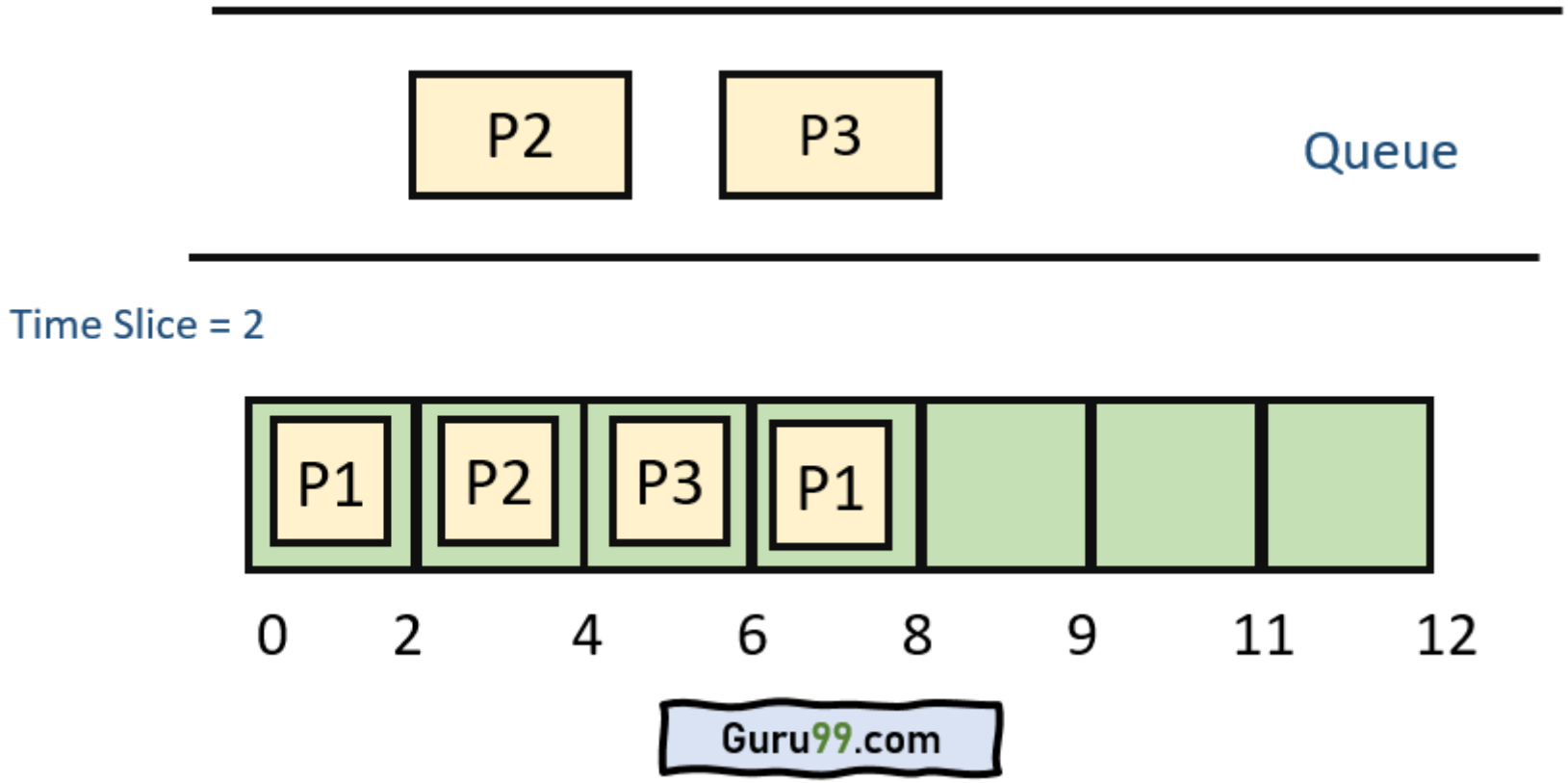
Step 2) At time =2, P1 is added to the end of the Queue and P2 starts executing



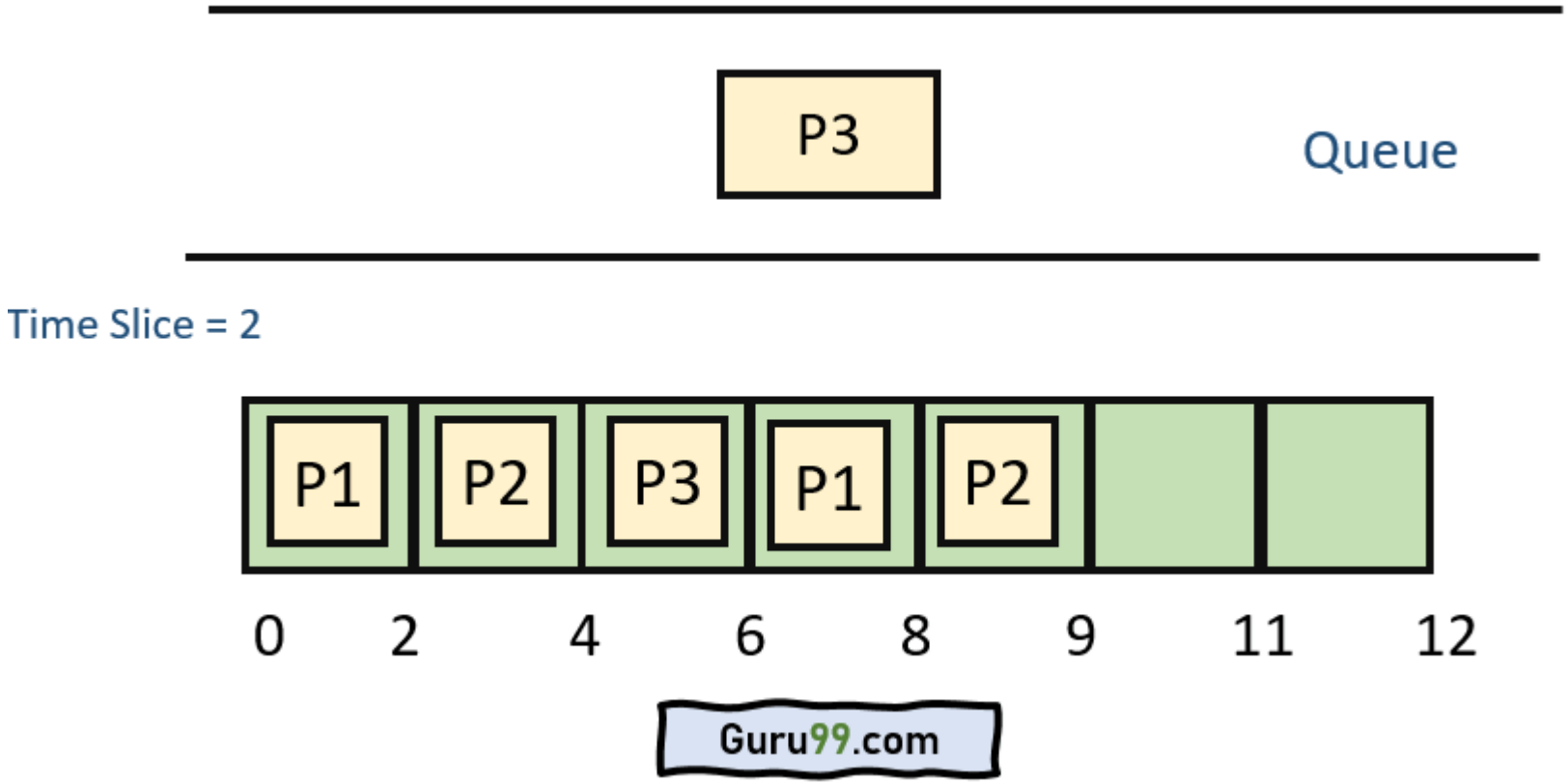
Step 3) At time=4 , P2 is preempted and add at the end of the queue. P3 starts executing.



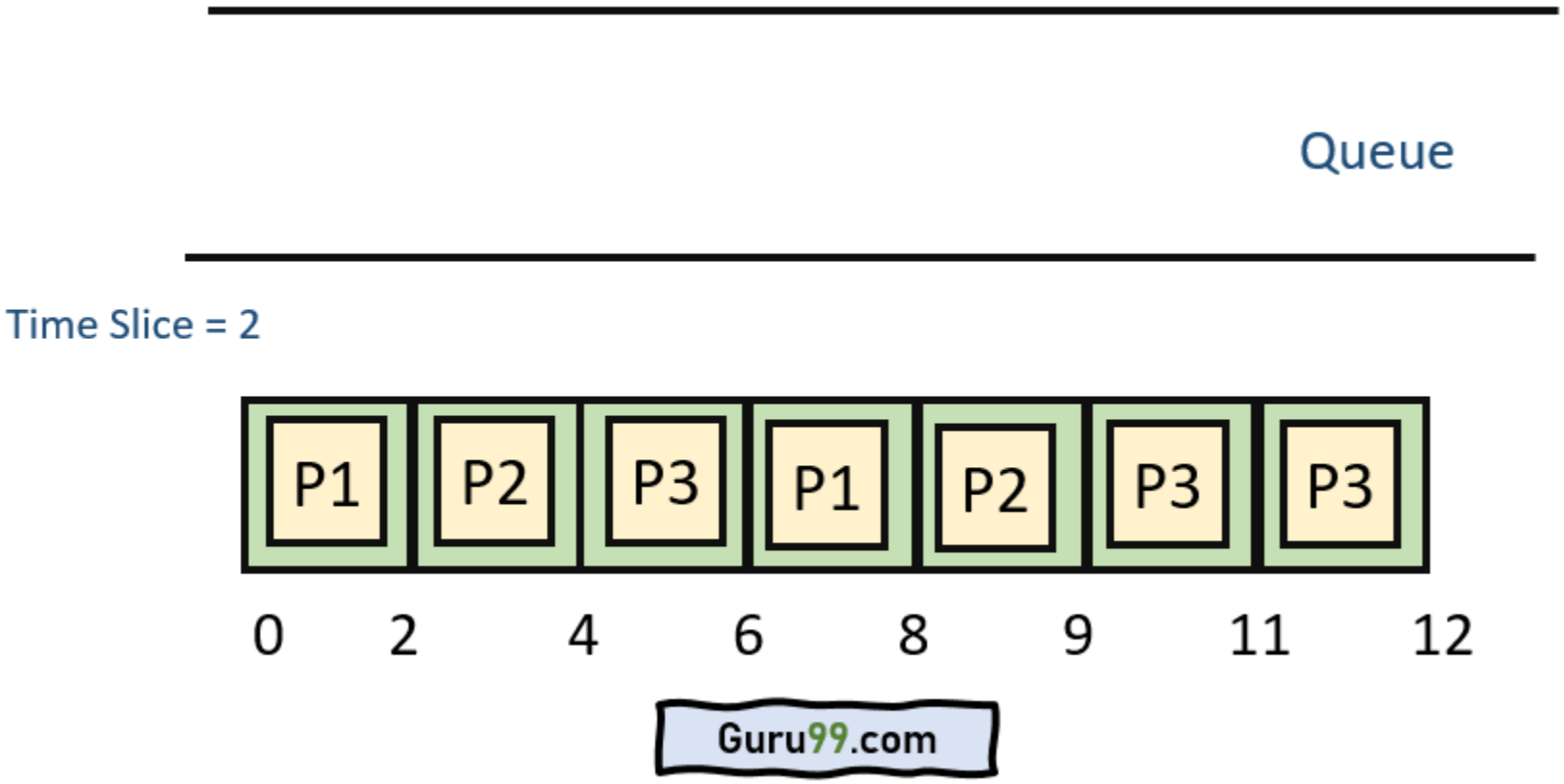
Step 4) At time=6 , P3 is preempted and add at the end of the queue. P1 starts executing.



Step 5) At time=8 , P1 has a burst time of 4. It has completed execution. P2 starts execution



Step 6) P2 has a burst time of 3. It has already executed for 2 interval. At time=9, P2 completes execution. Then, P3 starts execution till it completes.



Step 7) Let's calculate the average waiting time for above example.

Wait time

P1= 0+4= 4

P2= 2+4 = 6

P3= 4+3= 7

Advantages of Round Robin Scheduling

- It provides fairness by giving each process an equal share of the CPU time in a cyclic manner.
- Round robin scheduling typically has low latency because processes are served in a predictable order, ensuring that no process waits too long for CPU time.
- It's relatively easy to implement compared to other scheduling algorithms like priority scheduling or shortest job next (SJN).
- Round robin scheduling is well-suited for time-sharing systems where multiple users are accessing the system concurrently.

Dis-advantages of Round Robin Scheduling

- Processes with short bursts of CPU time may be delayed due to the fixed time slices allocated to each process.
- It might not be the most efficient scheduling algorithm when dealing with processes with vastly different CPU time
- In scenarios where there's high variability in process execution times, round robin may lead to poor performance due to frequent context switches.
- There's additional overhead associated with managing the ready queue and performing context switches at regular intervals

Worst Case Latency

- dt = Denote detection time when a task is brought into the list
- st = Denote switching time from one task to another
- et = Denote task execution time


THANK
YOU