

Exercise Series N°03 : Process Management

Exercise III.1: (Understanding Questions)

- Q1)** A CPU-scheduling algorithm determines an order for the execution of its scheduled processes. Given n processes to be scheduled on one processor, how many different schedules are possible? Give a formula in terms of n .
- Q2)** What is the role of the scheduler in an operating system?
- Q3)** Why is the SJF scheduling algorithm not really applicable?
- Q4)** Priority-based scheduling algorithms can lead to starvation (non-execution) of low-priority processes. How can this problem be avoided?
- Q5)** Explain the difference between preemptive and non-preemptive scheduling.

Exercise III.2: (Process Management With Switching Times)

Consider the following five processes.

Process	Arrival Time	CPU Execution (Burst) Time (ms)
P1	0	7
P2	1	4
P3	1	2
P4	2	2
P5	3	1

- Q1)** For each of the following scheduling algorithms :
- **FCFS** (First Come, First Served),
 - **SJF** (Shortest Job First),
 - **SRTF** (Shortest Remaining Time First), and
 - **Round Robin** (with a time quantum = 1 ms):
 - a) Show the scheduling order of the processes using a Gantt chart.
 - b) What is the turnaround time for each process? And average turnaround time?
 - c) What is the waiting time for each process? And Average waiting time?
- Q2)** If the context switching time is 0.5 ms, what is the average turnaround time for **SRTF** and **Round Robin** scheduling? What do you deduce from this?

Exercise III.3: (Process Management)

Consider a single-processor system and the four processes **P1**, **P2**, **P3** and **P4**, which perform computation and I/O with a disk according to the times given below.

The processes are available from the start, in this order : **P1** then **P2** then **P3** then **P4**.

We give: priority (**P1**) > priority (**P3**) > priority (**P2**) > priority (**P4**).

Process	P1	P2	P3	P4
CPU Burst Time (1 st)	3	4	2	7
I/O Burst Time (1 st)	7	3	3	
CPU Burst Time (2 nd)	2	2	2	
I/O Burst Time (2 nd)	1	1		
CPU Burst Time (3 rd)	1	1		

- Q1)** Consider that scheduling on the processor is : based on a **pre-emptive priority policy**: the process elected at time t is the highest-priority ready process. The order of I/O requests for the disk is always based on a **FCFS** policy.

Complete Appendix A

- Q2)** The processor scheduling **policy is unchanged**, but we now consider that the order of service of I/O requests for the disk is also based on process **priority**: the process starting an I/O is the one with the highest priority among those in the disk wait state. A started I/O operation cannot be pre-empted.

Complete Appendix B

- Q3)** Scheduling on the processor is based on a **Round Robin** policy with a quantum of **2 time units**. The order of arrival is assumed to be **P1** then **P2** then **P3** then **P4**. The order of service for disk I/O requests is assumed to be **FCFS** policy.

Complete Appendix C

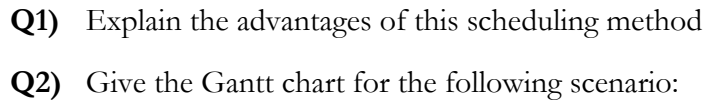
- Q4)** Based on your work above, fill in the table below giving both the waiting time and turnaround time for each process:

	Scheduling Algorithms					
	Algorithms 1 (Q1)		Algorithms 2 (Q2)		Algorithms 3 (Q3)	
Process	Waiting Time	Turnaround Time	Waiting Time	Turnaround Time	Waiting Time	Turnaround Time
P1						
P2						
P3						
P4						
Average						

Exercise III.4: (Multi-Level Scheduling)

Consider the following multi-level scheduling and feedback technique:

- **Queue 1** has the highest priority. A newly created process is placed in **Queue 1**, which is managed according to the **FCFS** algorithm.
- When a process in **Queue 1** obtains the processor, it is given a maximum time of **8 time units**; if it does not finish, it is moved to **Queue 2**.
- **Queue 2** is also managed according to **FCFS** scheduling, but each process is given a maximum time of **4 time units**. If the process does not finish, it is moved to **Queue 3**.
- **Queue 3** is managed using **Round Robin** scheduling, with a quantum equal to **2 time units**.



Q3) Based on your work above, fill in the table below.

Appendix A

[illegible]

Appendix B

[illegible]

Appendix C

[illegible]