Data Structure Projects

CPU scheduling deals with the problem of deciding which of the processes in the ready queue is to be allocated the CPU. There are different CPU scheduling algorithms that can be used.

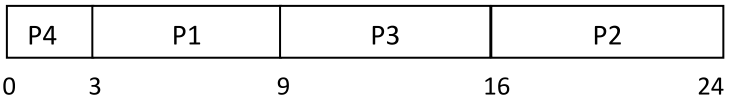
a) Shortest-Job-First Scheduling (SJF)

This algorithm associates with each process the "burst time", i.e. the time in which CPU can finish this process. When the CPU is available, it is assigned to the process that has the smallest CPU burst. If two processes have the same length, the FIFO scheduling is used.

Ex:

Process	Burst Time
P1	6
P2	8
P3	7
P4	3

They will be scheduled to CPU as follows:



Write a simulation program for the system described above. The program should display the following menu.

1. Add a process.
2. Serve a process.
3. How many waiting processes?
4. Display information of waiting processes (Input Order).
5. Display information of waiting processes (Served Order).
6. Display total CPU burst time..
7. Exit (Save in a file).

Hints: The process should have the following fields: process ID and burst time. The option "Display total CPU burst time" should display the burst time that the CPU has to take to serve all served processes. Option 4 should display the waiting processes in the order they were added to the system. Option 5 displays them in the order that they will be served.

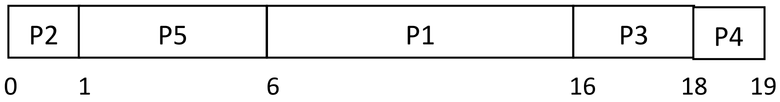
b) Priority Scheduling

A priority is associated with each process and the CPU is allocated to the process with the highest priority (small value). Equal priority processes are scheduled in FIFO order.

Ex:

Process	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

They will be scheduled to CPU as follows:



Write a simulation program for the system described above. The program should display the following menu.
1. Add a process.
2. Serve a process.
3. How many waiting processes?
4. Display information of waiting processes (Input Order).
5. Display information of waiting processes (Served Order).
6. Display total CPU burst time.
7. Exit (Save in a file).

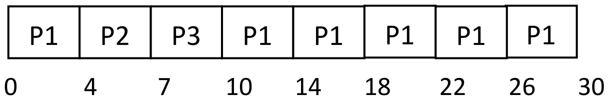
Hints: The process should have the following fields: process ID, burst time and priority. The option "Display total CPU burst time" should display the burst time that the CPU has to take to execute all served processes. Option 4 should display the waiting processes in the order they were added to the system. Option 5 displays them in the order that they will be served.

c) Round-Robin Scheduling (RR)

It is designed especially for time sharing systems. A small unit of time called a "time quantum" is defined. The ready queue is treated as circular queue. The CPU scheduler goes around the ready queue, allocating the CPU to each process for a time interval of up to a time quantum.

Ex:	Process	Burst Time
	P1	24
	P2	3
	P3	3

They will be scheduled to CPU as follows: (time quantum=3)



Write a simulation program for the system described above. The program should display the following menu.
1. Add a process.
2. Serve a process.
3. How many waiting processes?
4. Display information of waiting processes.
5. Display total CPU burst time.
6. Exit (Save in a file).

Hints: The process should have the following fields: process ID and burst time. The option "Serve a process" should display which process is served and how much time left for it (if it is not totally finished) but if it is totally executed, it must be served from the queue and display that this process is 'Totally done'. The option "Display total CPU burst time" should display the burst time that the CPU has to take. Option 4 should display the processes that are waiting in the queue and how much time left for it.