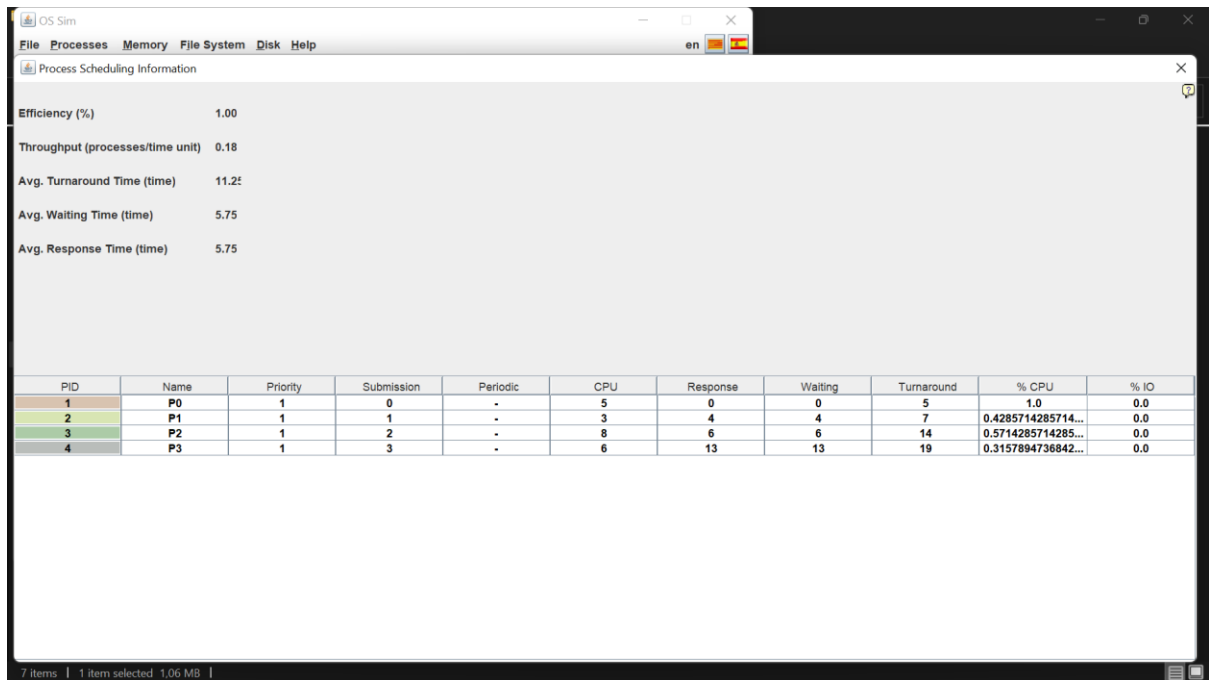


LAPORAN PRAKTIKUM SISTEM OPERASI MODUL 11

FCFS



The screenshot shows the 'Process Scheduling Information' window in OS Sim. It displays performance metrics for the FCFS scheduling algorithm. The metrics are: Efficiency (%) at 1.00, Throughput (processes/time unit) at 0.18, Avg. Turnaround Time (time) at 11.25, Avg. Waiting Time (time) at 5.75, and Avg. Response Time (time) at 5.75. Below these metrics is a table with 11 columns: PID, Name, Priority, Submission, Periodic, CPU, Response, Waiting, Turnaround, % CPU, and % IO. The table contains four rows of process data.

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
1	P0	1	0	-	5	0	0	5	1.0	0.0
2	P1	1	1	-	3	4	4	7	0.4285714285714...	0.0
3	P2	1	2	-	8	6	6	14	0.5714285714285...	0.0
4	P3	1	3	-	6	13	13	19	0.3157894736842...	0.0

Kesimpulan

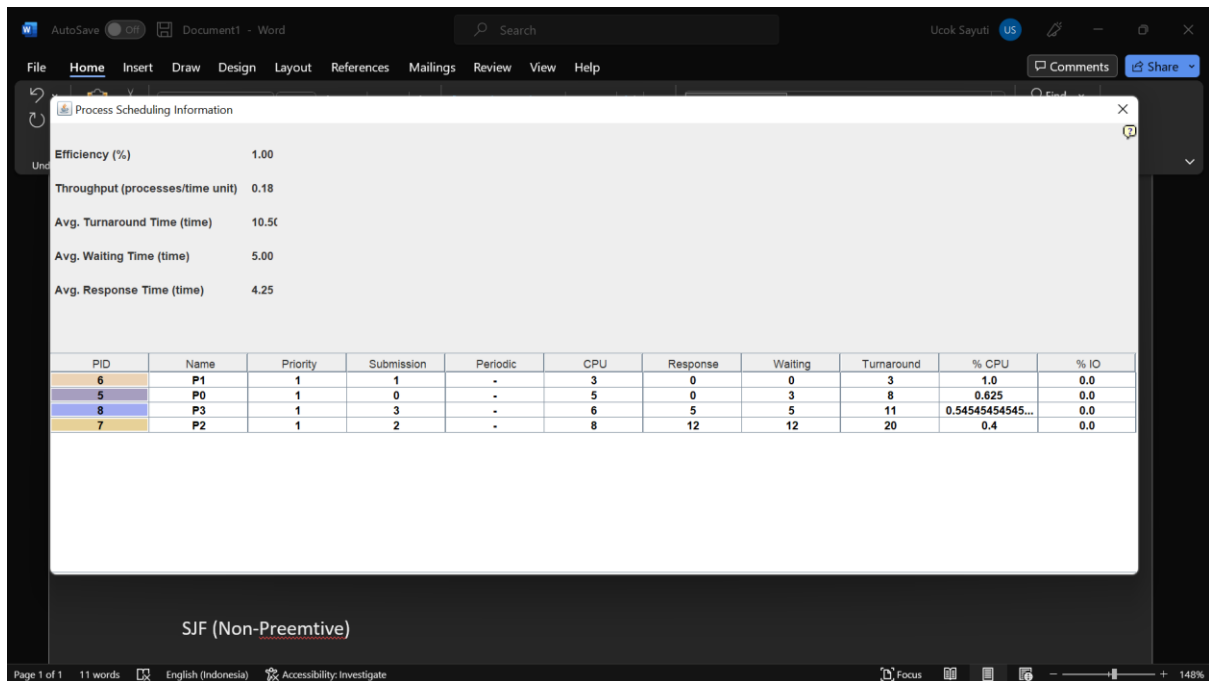
AWT (Average Waiting Time):

$$AWT = \sum WT / \sum Job$$

$$= 0 + 4 + 6 + 713 / 4 = 5,75$$

Jadi, rata-rata waktu yang dibutuhkan pada metode FCFS adalah **5,75 mili second**.

SJF (Preemptive)



Process Scheduling Information

Efficiency (%) 1.00

Throughput (processes/time unit) 0.18

Avg. Turnaround Time (time) 10.50

Avg. Waiting Time (time) 5.00

Avg. Response Time (time) 4.25

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
6	P1	1	1	-	3	0	0	3	1.0	0.0
5	P0	1	0	-	5	0	3	8	0.625	0.0
8	P3	1	3	-	6	5	5	11	0.5454545454...	0.0
7	P2	1	2	-	8	12	12	20	0.4	0.0

SJF (Non-Preemptive)

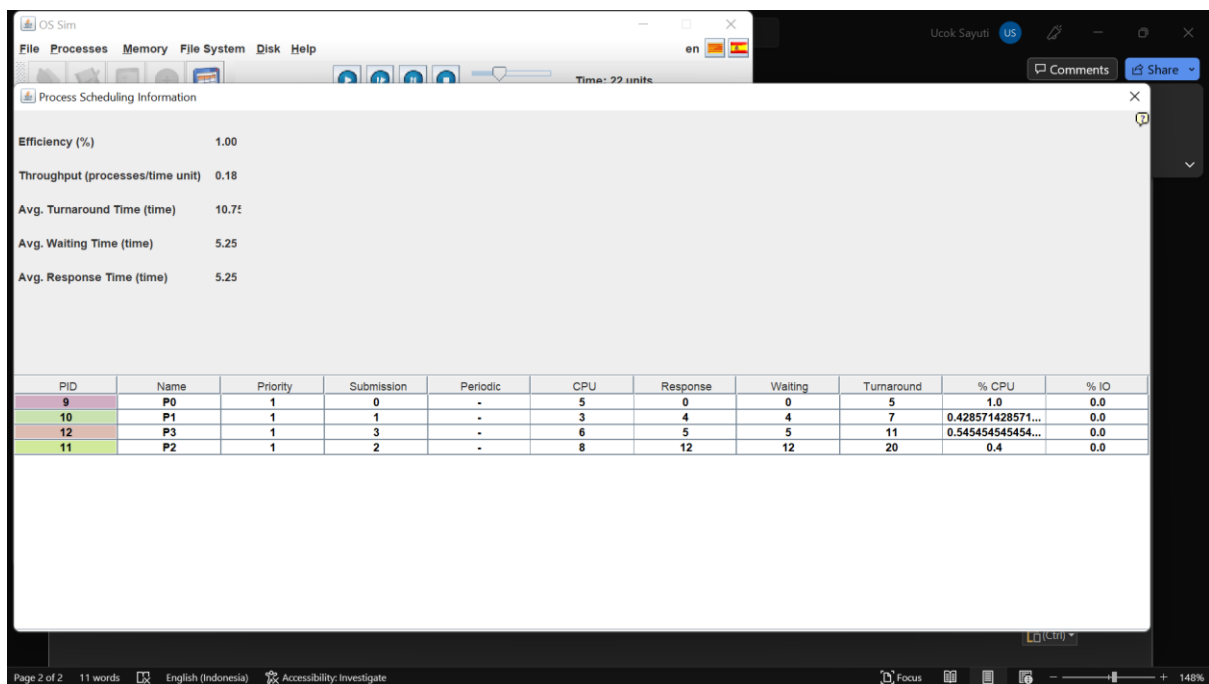
AWT (Average Waiting Time):

$$AWT = \sum WT / \sum Job$$

$$= 3 + 0 + 12 + 5 / 4 = 5,00$$

Jadi, rata-rata waktu yang dibutuhkan pada metode SJF preemptive adalah **5,00 mili second**.

SJF (Non-Preemptive)



The screenshot shows the 'Process Scheduling Information' window in OS Sim. It displays various performance metrics and a table of process execution details.

Summary Metrics:

- Efficiency (%): 1.00
- Throughput (processes/time unit): 0.18
- Avg. Turnaround Time (time): 10.75
- Avg. Waiting Time (time): 5.25
- Avg. Response Time (time): 5.25

Process Execution Table:

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
9	P0	1	0	-	5	0	0	5	1.0	0.0
10	P1	1	1	-	3	4	4	7	0.428571428571...	0.0
12	P3	1	3	-	6	5	5	11	0.545454545454...	0.0
11	P2	1	2	-	8	12	12	20	0.4	0.0

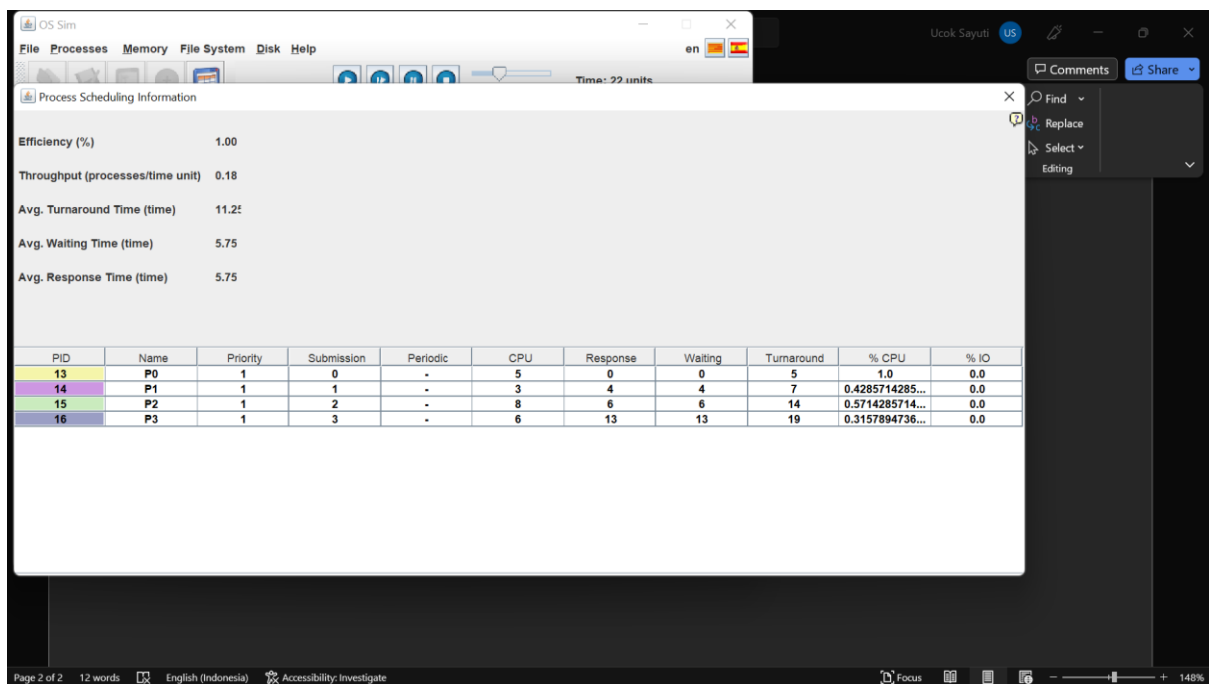
AWT (Average Waiting Time):

$$AWT = \sum WT / \sum Job$$

$$= 0 + 4 + 6 + 13 / 4 = 5,75$$

Jadi, rata-rata waktu yang dibutuhkan pada metode SJF non preemptive adalah **5,75 mili second**.

Priority



The screenshot shows the 'Process Scheduling Information' window in OS Sim. It displays various performance metrics and a table of process execution details. The metrics include Efficiency (1.00), Throughput (0.18), Avg. Turnaround Time (11.25), Avg. Waiting Time (5.75), and Avg. Response Time (5.75). The table lists four processes (P0, P1, P2, P3) with their respective submission times, priorities, and execution metrics.

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
13	P0	1	0	-	5	0	0	5	1.0	0.0
14	P1	1	1	-	3	4	4	7	0.4285714285...	0.0
15	P2	1	2	-	8	6	6	14	0.5714285714...	0.0
16	P3	1	3	-	6	13	13	19	0.3157894736...	0.0

Kesimpulan :

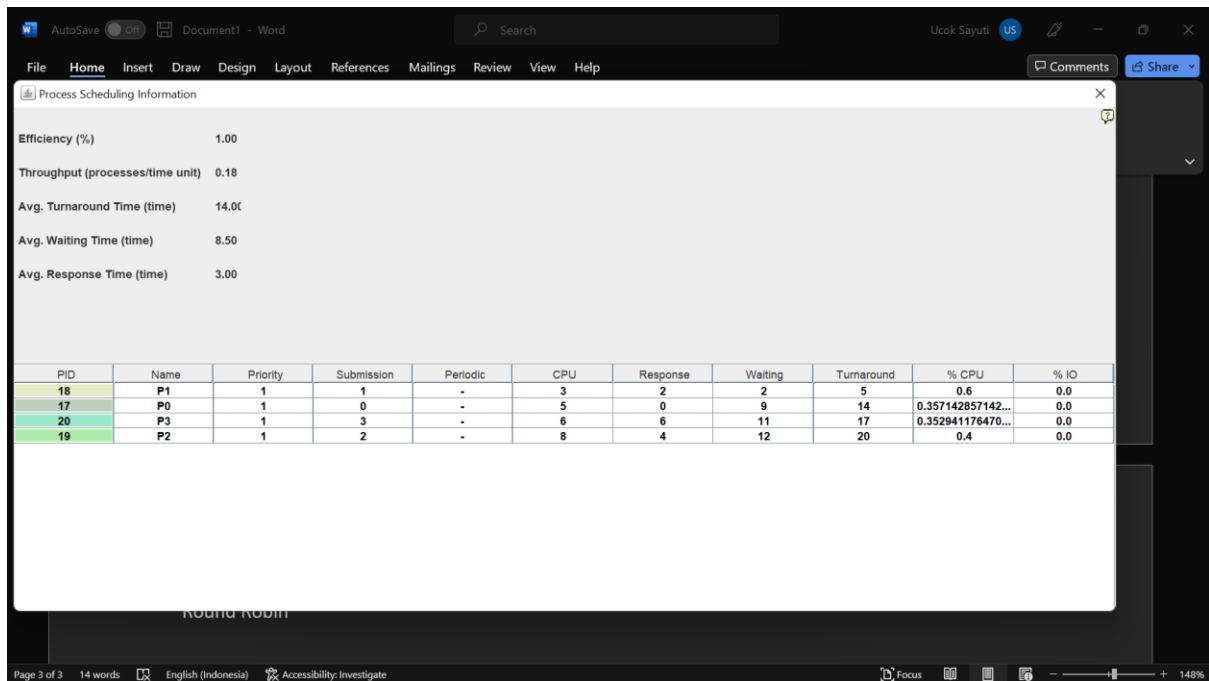
AWT (Average Waiting Time):

$$AWT = \sum WT / \sum Job$$

$$= 0 + 4 + 9 + 11 / 4 = 6,00$$

Jadi, rata-rata waktu yang dibutuhkan pada metode priority adalah **5,00 mili second**.

Round Robin



Process Scheduling Information

Efficiency (%) 1.00

Throughput (processes/time unit) 0.18

Avg. Turnaround Time (time) 14.00

Avg. Waiting Time (time) 8.50

Avg. Response Time (time) 3.00

PID	Name	Priority	Submission	Periodic	CPU	Response	Waiting	Turnaround	% CPU	% IO
18	P1	1	1	-	3	2	2	5	0.6	0.0
17	P0	1	0	-	5	0	9	14	0.357142857142...	0.0
20	P3	1	3	-	6	6	11	17	0.352941176470...	0.0
19	P2	1	2	-	8	4	12	20	0.4	0.0

Kesimpulan :

AWT (Average Waiting Time):

$$AWT = \sum WT / \sum \text{Job} = 9 + 2 + 12 + 11 / 4 = 8,50 \text{ ms}$$

Jadi, rata-rata waktu yang dibutuhkan pada metode Penjadwalan Round Robin (RR) dengan quantum time= 3 ms adalah **8,50 mili second**.