CSC 2405 – CPU Scheduling Exercises

Exercise 1 [Predicting lengths of CPU bursts]

Suppose that a process is given a default expected burst length of 5 time units when first created. Consider now a process P whose actual CPU burst lengths are 10, 10, 10, 1, 1, 1, 1 (although this information is not known in advance to any scheduling algorithm).

Assuming that the exponential averaging algorithm for predicting CPU time bursts uses $\alpha = 0.5$, calculate the expected burst times e(2), e(3), …, e(7) for this process (note that e(1) = 5).

Recall that the exponential averaging formula is $e(n+1) = (1-\alpha)*e(n)+\alpha*t(n)$

Exercise 2 [CPU Scheduling, no I/O]

Consider seven processes P_1, P_2, …, P_7 with arrival times and CPU burst times as follows:

Process	P1	P2	P3	P4	P5	P6	P7
Arrival time	2-ε	4-ε	5-ε	7-ε	9-ε	15-ε	16-ε
CPU burst time	3	2	1	4	2	6	8

Here "2-ε" indicates that P1 has arrived just before time unit 2, and similarly for the others. Assume that, when joining the Ready Queue, (new or existing) processes always get appended at the end of the queue.

Draw a chart that illustrates the execution of these processes using the specified scheduling algorithm. Calculate the average turnaround time and waiting time.

(a) FCFS (First Come First Served)

P1	P2	P3	P4	P5		P6	P7
5	7	8	12	14	15	21	29

Time	CPU Use	Ready Queue (at end of time slot)	Event
2-5	P1	P2 , P3	P2 arrives at 4-ε, P3 arrives at 5-ε
5-7	P2	P3, P4	P4 arrives at 7-ε
7-8	P3	P4	
8-12	P4	P5	P5 arrives at 9-ε
12-14	P5	Empty	
14-15	Idle	P6	P6 arrives at 15-ε
15-21	P6	P7	P7 arrives at 16-ε
21-29	P7	Empty	

Process	P1	P2	P3	P4	P5	P6	P7
End time	5	7	8	12	14	21	29
Turnaround time	3	3	3	5	5	6	13
Waiting time	0	1	2	1	3	0	5

Average turnaround time = (3+3+3+5+5+6+13)/7
Average waiting time = (0+1+2+1+3+0+5)/7

(b) RR (Round Robin, time quantum = 1)

Time	CPU Use	Ready Queue (at end of time slot)	Event
2-4	P1	P2	P2 arrives at 4-ε
4-5	P2	P1, P3	P3 arrives at 5-ε
5-6	P1	P3, P2	P1 done
6-7	P3	P2, P4	P4 arrives at 7-ε; P3 done
7-8	P2	P4	P2 done
8-9	P4	P5	P5 arrives at 9-ε
9-10	P5	P4	
10-11	P4	P5	
11-12	P5	P4	P5 done
12-14	P4	Empty	P4 done
14-15	Idle	P6	P6 arrives at 15-ε
15-16	P6	P7	P7 arrives at 16-ε
16-17	P7	P6	
17-18	P6	P7	
…			
25-26	P6	P7	
26-29	P7	Empty	

P1	P1	P2	P1	P3	P2	P4	P5	P4	P5	P4	P4		P6	P7	…	P6	P7	P7	P7
3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	…	26	27	28	29

Process	P1	P2	P3	P4	P5	P6	P7
End time	6	8	7	14	12	26	29
Turnaround time	4	4	2	7	3	11	13
Waiting time	1	2	1	3	1	5	5

Average turnaround time = (4+4+2+7+3+11+13)/7
Average waiting time = (1+2+1+3+1+5+5)/7

(c) SPN (Shortest Process Next, non-preemptive)

(d) PSPN (Shortest Process Next, preemptive)

<u>Exercise 3</u> [CPU Scheduling, parallel I/O]

The Ready queue of an operating system at a particular time instance is as follows:

Process	Next CPU burst (in milliseconds)
P1	2
P2	3
P3	7
P4	18

The behavior of each process (if it were to use the CPU exclusively) is as follows: it runs for the CPU burst given, then requests an I/O operation that takes 10 milliseconds, then runs for another CPU burst of equal duration to its first CPU burst and then terminates.

However, the four processes must share the CPU. Assume that the I/O operations can proceed in parallel.

a) Draw a chart showing the execution of these processes under RR, with time quantum = 2

b) Draw a chart showing the execution of these processes under preemptive PSPN. For each process, calculate the total missed time (time spent in the Ready queue).

Exercise 4 [CPU Scheduling, sequential I/O]

Consider a system running two CPU-bound jobs C1 and C2, and four I/O-bound jobs O1, O2, O3 and O4. Each I/O bound task issues an I/O operation once every 1 millisecond of CPU. Each I/O operation takes 4 milliseconds. Assume that there is only one I/O device (so multiple I/O requests may have to queue). Assume that the context switch takes 0.2 milliseconds.

Assume that each CPU-bound requires 10 milliseconds of CPU to complete and each I/O-bound task requires 2 milliseconds of CPU time. Assume that all jobs are in the Ready queue at time 0, in the order C1, C2, O1, O2, O3, O4 (front to back).

Draw a scheduling chart to show how the I/O and CPU are allocated and compute the average turnaround times for the CPU-bound and I/O bound tasks, for each of the following 2 cases. Use a table similar to the one below to keep track of all events and Ready / Blocked (I/O) queues.

a) The system uses Round Robin scheduling with time slice (quantum) of 10 milliseconds

Time	CPU Use	Ready Queue (end of time slot)	I/O Use	I/O Queue
0-10	C1	C2, O1, O2, O3, O4	-	-
10-10.2	Context Switch	O1, O2, O3, O4	-	-
10.2-20.2	C2	same	-	-
20.2-20.4	Context Switch	O2, O3, O4	-	-
20.4-21.4	O1	same	-	-
21.4-21.6	Context Switch	O3, O4	O1	-

(continue from here …)

b) The system uses Round Robin scheduling with time slice (quantum) of 5 milliseconds