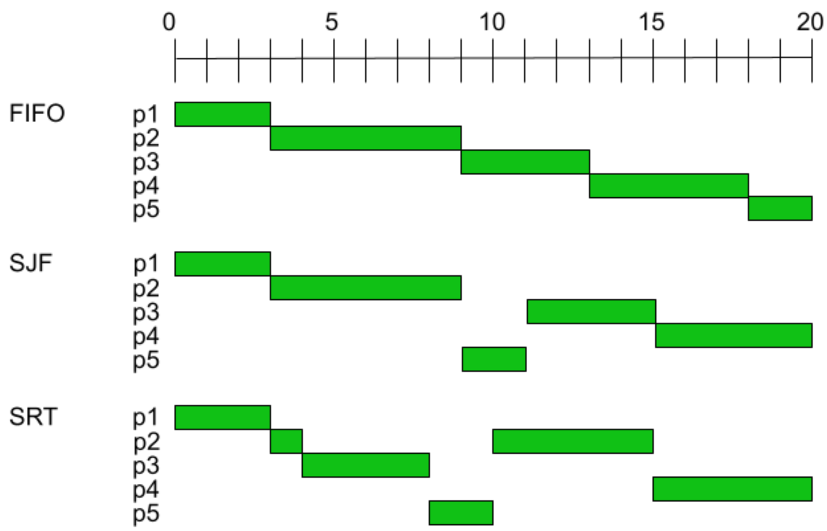
Name: Jeremiah Webb
Class: CS420
Date: 17th November 2022

3.2.1
(a)
For the 5 processes described below, draw a timing diagram showing when each process executes under FIFO, SJF, and SRT.



(b)
Determine the ATT for each scheduling algorithm for the 5 processes.

Process	p1	p2	p3	p4	p5	ATT
FIFO: Turnaround times	3	7	9	12	12	8.60
SJF: Turnaround times	3	7	11	14	3	7.60
SRT: Turnaround times	3	13	4	14	2	7.20

3.2.2 (a)
Starting at time 0, a new process p of length 3 arrives every 4 time units.
Starting at time 1, a new process q of length 1 arrives every 4 time units.
Determine the ATT under FIFO, SJF, and SRT.

	p	q	ATT
	waiting+running	waiting+running	
FIFO, SJF	0+3	2+1	6/2 = 3
SRT	1+3	0+1	5/2 = 2.5

3.2.3
(a)

i	0	1	2	3	4	5
T_i	7	5	6	15	15	15
S_i	7	7	5.4	5.88	13.176	14.635

(b)

i	0	1	2	3	4	5
T_i	7	5	6	15	15	15
S_i	7	7	6	6	10.5	12.75

3.2.4
(a)

$S_5 = 0.5 * T_4 + 0.5 * S_4$ Thus T_4 contributes 50% to S_5
$= 0.5 * T_4 + 0.5 * (0.5 * T_3 + 0.5 * S_3)$
$= 0.5 * T_4 + 0.25 * T_3 + 0.25 * S_3$ Thus T_3 contributes 25% to S_5
$= 0.5 * T_4 + 0.25 * T_3 + 0.25(0.5 * T_2 + 0.5 * S_2)$
$= 0.5 * T_4 + 0.25 * T_3 + 0.125 * T_2 + 0.125 * S_2$ Thus T_2 contributes 12.5% to S_5
$= 0.5 * T_4 + 0.25 * T_3 + 0.125 * T_2 + 0.125(0.5 * T_1 + 0.5 * S_1)$
$= 0.5 * T_4 + 0.25 * T_3 + 0.125 * T_2 + 0.0625 * T_1 + 0.0625 * S_1$ Thus T_1 contributes 6.25% to S_5.

3.3.1
(a) Determine the average turnaround time, ATT, when the quantum is Q = 1 time unit.

Time	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
p1	x			x			x			x		x		x	x	x
p2		x			x			x								
p3			x			x			x		x		x			

ATT = (16 + 8 + 13)/3 = 12.33

(b)Determine the average turnaround time, ATT, when the quantum is Q = 3 time units.

Time	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
p1	x	x	x							x	x	x			x	x
p2				x	x	x										
p3							x	x	x				x	x		

ATT = (16 + 6 + 14)/3 = 12

3.3.2 (a) How long will the execution take on a machine with n CPUs?
T ms

(b)How long will the execution take on a single CPU machine when the context switch overhead is zero?

n*t tms

(c) How long will the execution take on a single CPU machine when:

- the length of the time quantum is Q ms
- the time to perform each context switch is S ms

# of context switches: n*T/Q, total execution: n*T + (n*T/Q)*S ms

(d) Repeat the previous calculation using n=5, T=10,000, Q=100, S=10.

5*10,000 + (5*10,000/100)*10 = 55,000 ms.

3.3.3
(a) What should be the quantum size Q such that the gap between the end of one quantum and the start of the next quantum of any process does not exceed M ms?
Between the end of one quantum and the start of the next quantum, n - 1 processes will each execute one quantum: (n - 1)Q.
Additionally, n context switches will be needed for the interrupted process to restart: nS.
The sum of the two times must not exceed the limit M, i.e., $(n - 1)Q + nS \le M$. The largest Q that satisfies this condition is $Q = (M - nS)/(n-1)$.

(b) For n = 5, S = 10, and M = 450, M = 90, M = 50, determine:

- The corresponding values of Q
- The percentage of CPU time wasted on context switching

M	Q	% wasted
450	100	100*10/(100 + 10) = 9.09%
90	10	100*10/(10 + 10) = 50%
50	0	100*10/(0 + 10) = 100%

3.3.4
(a) T < Q

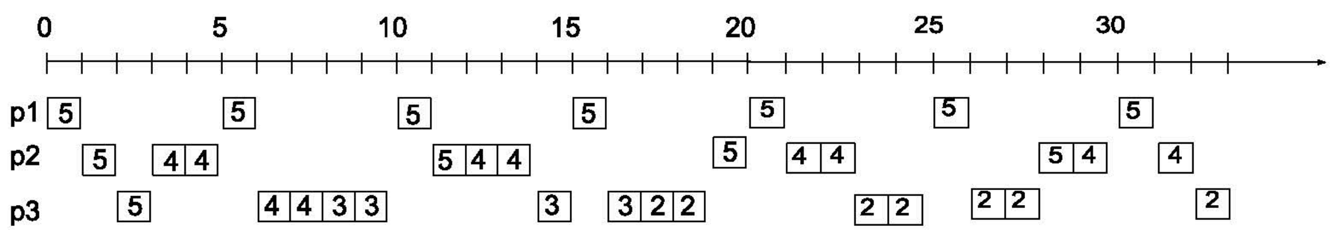
When T < Q, the quantum never expires. The execution repeats the timing sequence: T, S, T, S, ... Thus for every T ms of execution, T + S ms of CPU time are needed. The fraction of wasted CPU time is S/(T + S).

(b) T >> Q
When T >> Q, each execution of T needs many quanta to complete. Each quantum is followed by a context switch. The fraction of wasted CPU time is S/(Q + S).

(c)Q approaches 0
When Q approaches 0, the value of S/(Q + S) approaches 1 (overhead approaches 100%).

(d)Under what condition will the wasted fraction of CPU time be 50%?
50% overhead means S/(Q + S) = 0.5, which is true when Q = S.

3.3.5 (a)Draw a timing diagram for the first 33 ms. On each of the 3 lines (one per process) show when the process is running and at which priority level.



(b) Determine the ATT for each process.
P1 – ATT = 1
P2 – ATT = 4
P3 – ATT = 32

3.4.1 (a) Determine if a feasible schedule exists.
Determine if a feasible schedule exists.

5/20 + 10/100 + 42/120 = 0.25 + 0.1 + 0.35 = 0.7

0.7 < 1 and thus a feasible schedule exists.

(b)
5/20 + 10/100 + 42/120 = 0.25 + 0.1 + 0.35 = 0.7

0.7 < 1 and thus a feasible schedule exists.

(c)

The CPU fraction used by the 3 processes p1 through p3 is already 0.7.
Adding any more processes to the mix would likely result in an infeasible schedule.

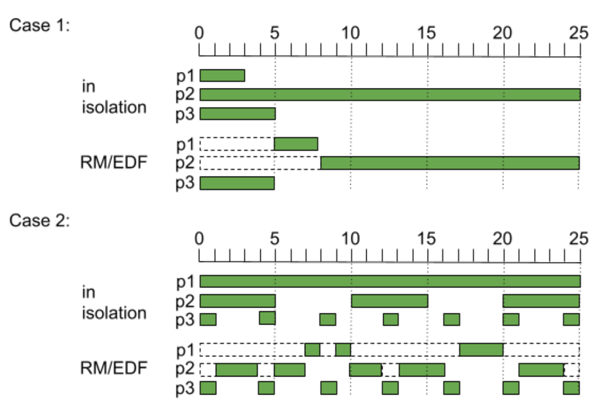
3.4.2
(a)
For each case, determine if a feasible schedule is likely to be generated by:

- RM
- EDF

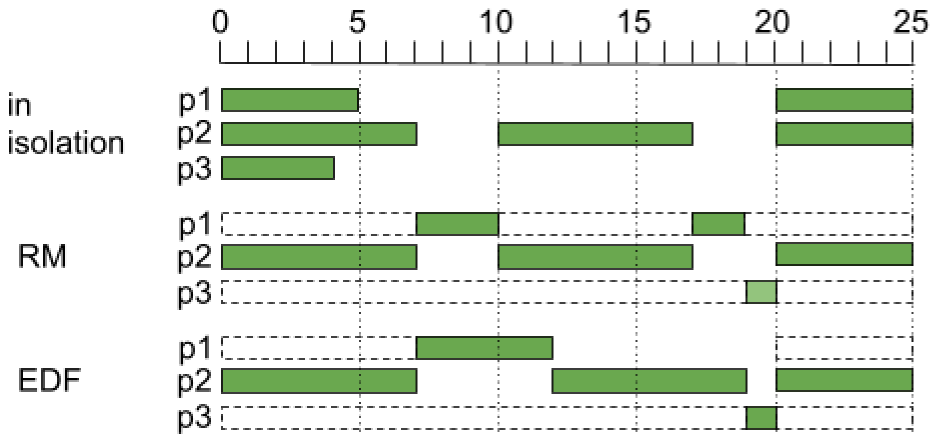
Case 1: Both RM & EDF. U = 3/50 + 70/1000 + 5/40 = 0.255
Case 2: Neither RM Nor EDF. U = 5/50 + 5/10 + 1/4 = 1.05
Case 3: EDF will produce feasible schedule. RM will most likely not produce a feasible schedule.
U = 5/20 + 7/10 + 4/100 = 0.99

(b)For each case, determine if a feasible schedule is likely to be generated by:

- RM
- EDF



Case 3:



3.5.1

(a)

Show at which priority level the process p is executing during each of the 18Q time units.

Running interval	Priority levels	Explanation
1st 3Q	4, 3, 3	p starts at level 4, runs for 1Q and drops to level 3 for the remaining 2Q units.
Next Q	2	p drops to level 2 and runs for 1Q to execute the blocking operation.
Next 7Q	3, 3, 2, 2, 2, 2, 1	p re-enters at level 3 and runs for 2Q, then for 4Q at level 2, then drops to level 1 for 1Q.
Next Q	1	p continues running at level 1 for 1Q to execute the blocking operation.
Next 3Q	2, 2, 2	p re-enters at level 2 and completes all 3Q at the same level 2.
Next Q	2	p continues running at level 2 for 1Q to execute the blocking operation.
Next 2Q	3, 3	p re-enters at level 3 and completes both units at the same level 3.