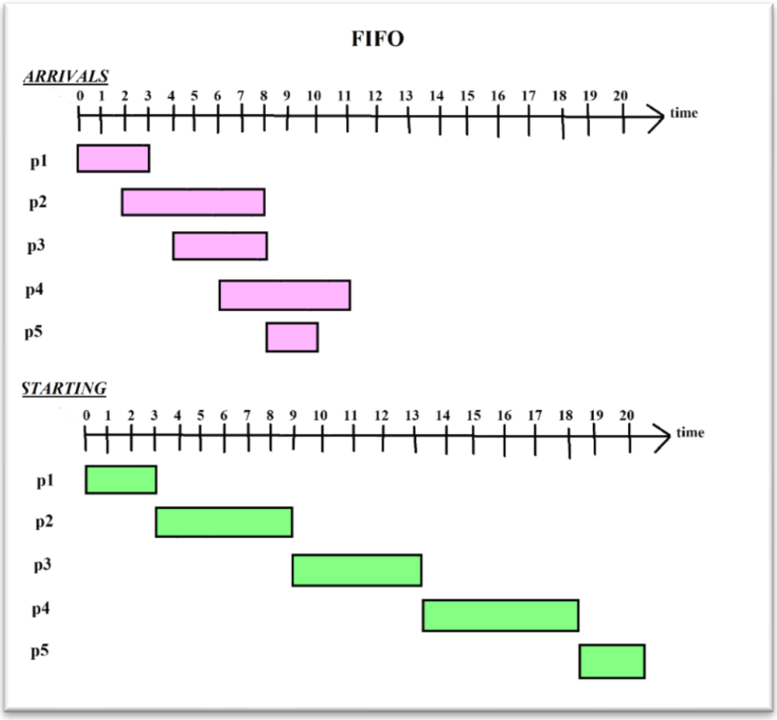
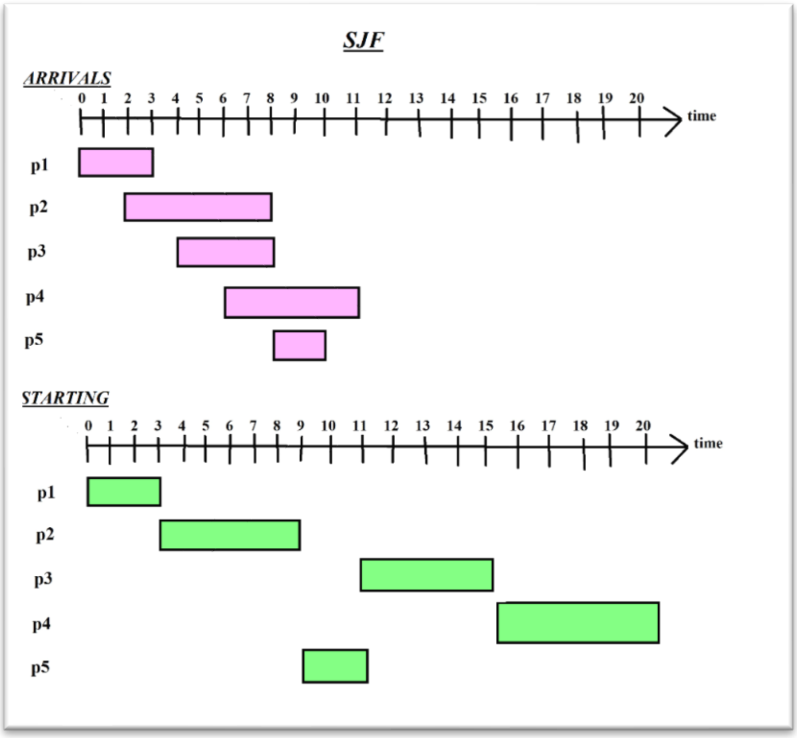
HÁSKÓLINN Í REYKJAVÍK
REYKJAVIK UNIVERSITY

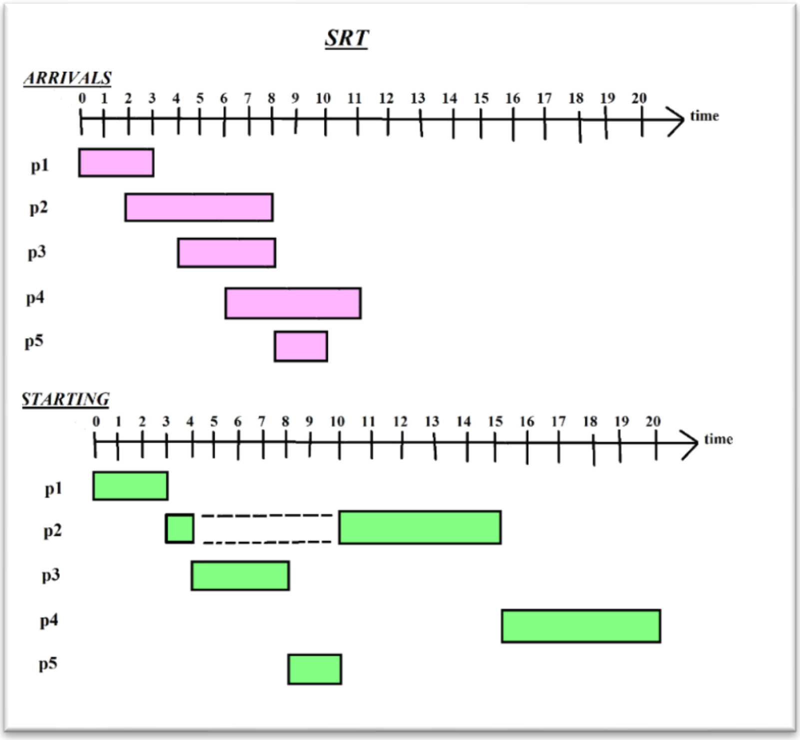
# *Benjamín Aage B. Birgisson*

## Exercise 3.2.1:

(a)






(b)

| | p1 | p2 | p3 | p4 | p5 | ATT |
|---|---|---|---|---|---|---|
| FIFO | 0+3 | 1+6 | 5+4 | 7+5 | 10+2 | 43 / 5 = 8.6 |
| SJF | 0+3 | 1+6 | 7+4 | 9+5 | 1+2 | 38 / 5 = 7.6 |
| SRT | 0+3 | 6+6 | 0+4 | 9+5 | 0+2 | 35 / 5 = 7.0 |

## Exercise 3.2.2:

(a)

| | p | q | ATT |
|---|---|---|---|
| FIFO | 0+3 | 2+1 | 6 / 2 = 3.0 |
| SJF | 0+3 | 2+1 | 6 / 2 = 3.0 |
| SRT | 1+3 | 0+1 | 5 / 2 = 2.5 |

# Exercise 3.2.4:

(a)

- As next prediction is based equally on $T_n$ and $S_n$, and $T_n$ is equal to the four previous T's, each $T_i$ contribution to $S_5$ would thereby be (1 / 4 =) 0.25.

---

# Exercise 3.3.1:

(a)

| | p1 | p2 | p3 | ATT |
|---|---|---|---|---|
| **RR** | 8+8 | 5+3 | 8+5 | 37 / 3 = 12.33 |

(b)

| | p1 | p2 | p3 | ATT |
|---|---|---|---|---|
| **RR** | 8+8 | 3+3 | 9+5 | 36 / 3 = 12.0 |

# Exercise 3.3.2:

(a)

- T ms

(b)

- n * T ms

(c)

i. T ms / Q ms
ii. T ms + S ms

(d)

- Total execution time : T ms * n
  = 10,000 * 5
  = 50,000 ms

iii. Lengt of the time quantum is Q ms : (T ms * n) / Q
  = 50,000 / 100
  = 500

iv. Time to perfom with S ms overhead: S / Q
  = 10 / 500
  = 0.02

v. Total execution time : 50,000 ms + (50,000 * 0.02)
  = 50,000 ms + 1,000 ms
  = 51,000 ms

## Exercise 3.3.3:

(a)

- n processes
- Q time quantum
- T ms of CPU time to complete each process
- S ms context switching
- M upper bound for the time between the end of one quantum and the start of the next quantum

➔ M > Q = n * T + S
➔ M > (n * T + S) / Q

(b)

## Exercise 3.3.4:

(a) T < Q
- The "Time quantum (Q)" is larger than the "Average total CPU time (T)".
- When multiple processes time-share the CPU, the response time increases with the number of processes, the length of the time quantum (Q), and the time it takes to perform a context switch.

(b) T >> Q
- The "Average total CPU time (T)" is much larger than the "Time quantum (Q).
- When the average total CPU time is much larger than the time quantum (Q), means that the context switch is again and again occurring, without the need of total time for each process. So, increased time waste on switching between processes.

(c) Q approaches 0

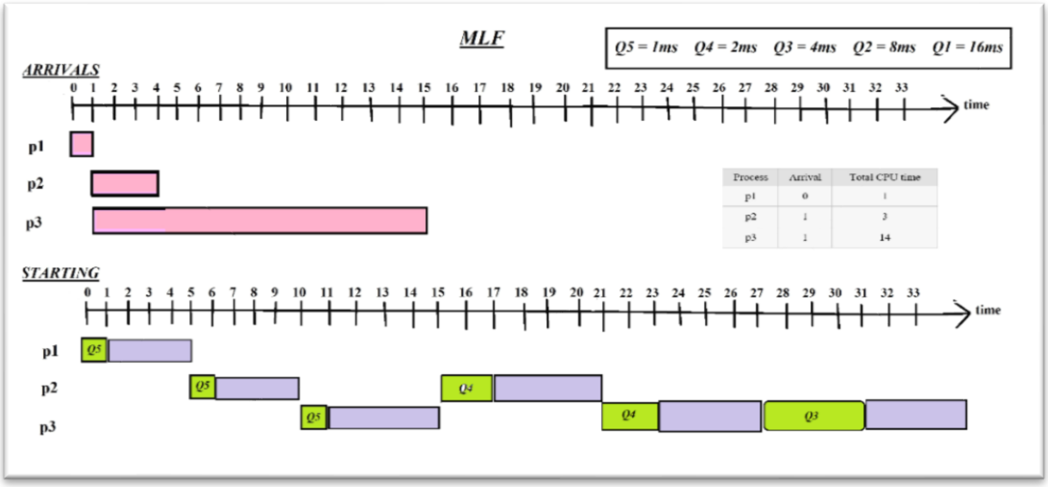
- The time wasted on context switching is very large, as Q approaches zero, because the time a process is allowed to use the CPU is very little (thereby, increasing the time wasted in switching between processes constantly).

(d) Under what condition will the wasted fraction of CPU time be 50%?

- When two processes time-share the CPU and context switching overhead is negligible, the CPU is divided evenly between the two processes and each process gets 50% of the CPU time.

## Exercise 3.3.5:

(a)




(b)

| | p1 | p2 | p3 | ATT |
|---|---|---|---|---|
| MLF | 0+1 | 9+3 | 16+7 | 36 / 3 = 12.0 |

---

HÁSKÓLINN Í REYKJAVÍK
REYKJAVIK UNIVERSITY

## Exercise 3.4.1:

(a)

- Given that T equals „Total CPU time“, and D equals „Period“, then:

U = 5/20 + 10/100 + 42/120 = 0,25 + 0,1 + 0,35 = 0,7

As U <= 1 EDF produces a feasible scheduled, and for RM then it is likely that it won't produce a feasible schedule as U is not less than approximately 0,7 (it could however, in some cases do so however).

(b)

**U = (3/20) * 6 = 0,9 <= 1**

There could be six processes run concurrently using EDF, as you have the delay period of each being 20 and 6 * 3 = 18, therefore you can run the first process between 0-3 and then again at time period 20-23 etc. keeping all deadlines intact (CPU is utilized 90%).

(c)

**U = (3/20) * 4 = 0,6 <= 0,7**

There could be four processes run concurrently using RM, guaranteeing a feasible schedule (all deadlines are met).

---------------------------------------------------------------------------------------

## Exercise 3.5.1:

(a)

*p runs for 3Q starting at level 4:*

| | |
|---|---|
| Priority level: 4 | Q time units: 1Q |
| Priority level: 3 | Q time units: 2Q |
| Priority level: 2 | |

*p blocks on keyboard input:*

Priority level: 2 + 2 = 4

*p wakes up and continues running for 7Q:*

| | |
|---|---|
| Priority level: 4 | Q time units: 1Q |
| Priority level: 3 | Q time units: 2Q |
| Priority level: 2 | Q time units: 4Q |
| Priority level: 1 | |

*p blocks on disk input:*

Priority level: 1 + 1 = 2

*p wakes up and continues running for 3Q:*

Priority level: 2 Q time units: 3Q

*p blocks on keyboard input:*

Priority level: 2 + 2 = 4

*p wakes up and continues running for 2Q:*

Priority level: 4 Q time units: 1Q

Priority level: 3 Q time units: 1Q

---