Assignment 2

Part 1

a) Possible events that can make a process abandon the use of CPU:

* I/O or event wait: This usually happens when the process is waiting for an I/O. During this waiting time, the CPU is given to the next process in line until the event occurs.

* Release/exit: This happens when the process is done with the CPU. The next process in line now has control of the CPU.

* Interrupt/Timeout: This usually happens when the process takes too much time while using the CPU. In fact, there is a timer allocated everytime a process starts running. When the process goes over the time limit, the next process in line takes its place and start using the CPU.

b) Difference between user-level thread and kernel-level thread:

↳ With the user-level thread, all thread management is done by the application by using a thread library in the user space while with the kernel-level thread, all thread management is done by the kernel.

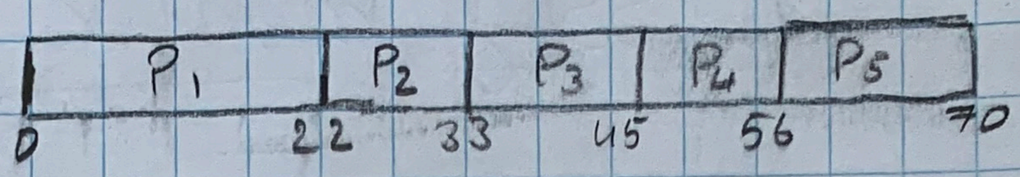
↳ With the user-level thread, scheduling can be application specific which can be faster than a kernel-level thread.

↳ With a kernel-level thread, multithreading is allowed and a system call only blocks one thread compared to the user-level thread.

In summary, the kernel-level thread includes more functionalities but is slower compared to the user-level thread.

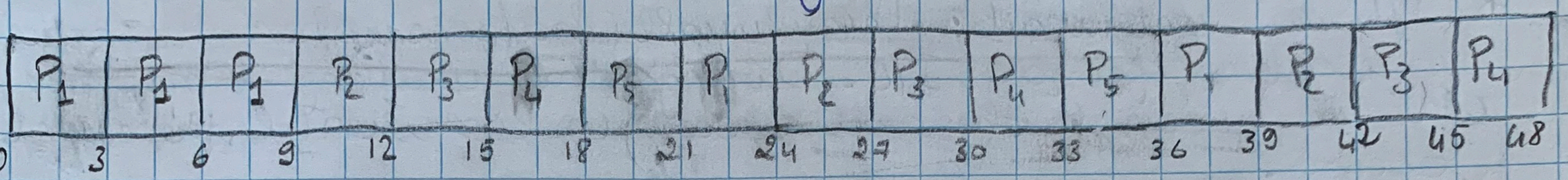
c) Compute the turnaround time:

i) FCFS

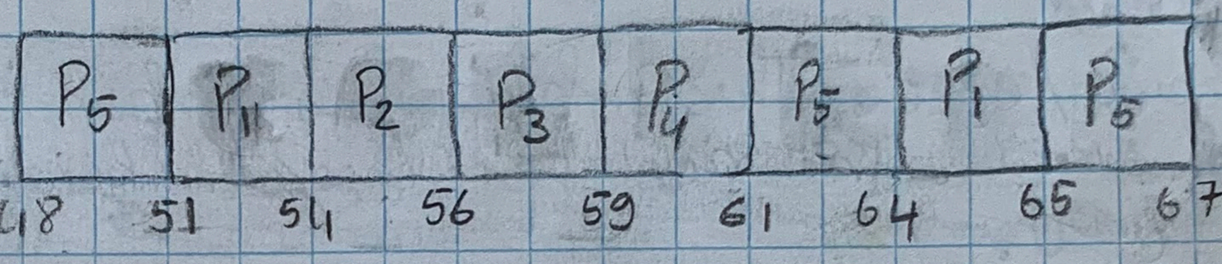


$$\text{Turnaround Time} = \frac{22 + 11 + 12 + 11 + 14}{5} = 14 \text{ sec}$$

ii) Round Robin (time slice of 3 sec)



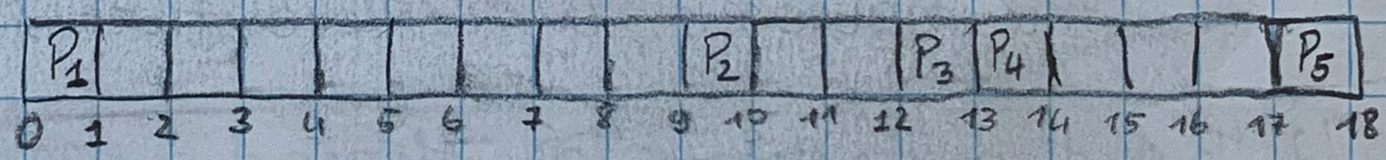
$P_1 \rightarrow 13 \rightarrow 10 \rightarrow 7 \rightarrow 4 \rightarrow 1 \rightarrow 0$
$P_2 \rightarrow 8 \rightarrow 5 \rightarrow 2 \rightarrow 0$
$P_3 \rightarrow 9 \rightarrow 6 \rightarrow 3 \rightarrow 0$
$P_4 \rightarrow 8 \rightarrow 5 \rightarrow 2 \rightarrow 0$
$P_5 \rightarrow 11 \rightarrow 8 \rightarrow 5 \rightarrow 2 \rightarrow 0$.



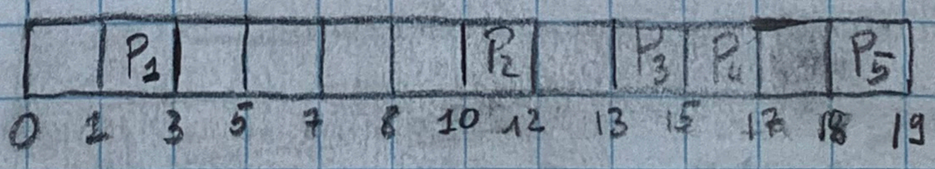
$$\text{Turnaround Time} = \frac{65 + 56 + 59 + 61 + 67}{5} = 61.6 \text{ sec}$$

iii) Multiple queue with feedback:

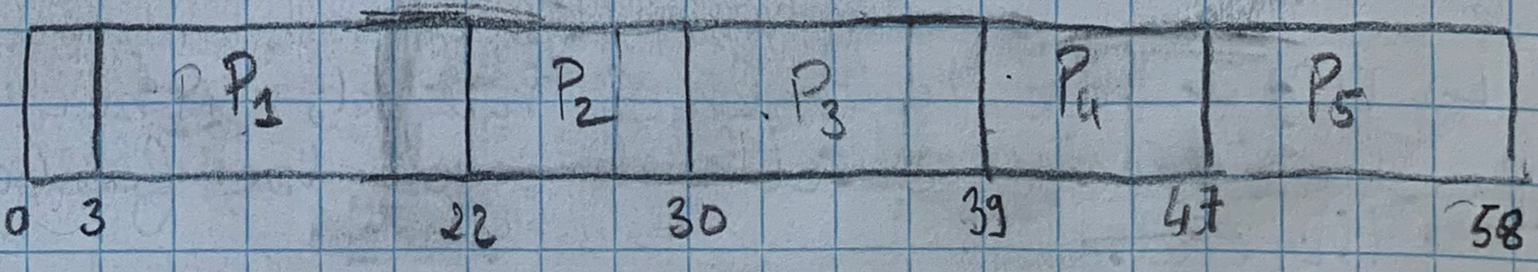
High-Priority Queue:



Mid-Priority Queue:



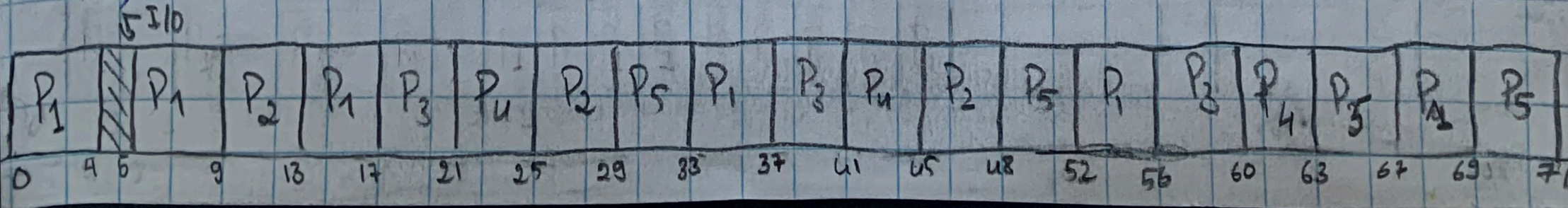
Low Priority Queue:



$$\text{Turnaroud Time} = \frac{22 + 30 + 39 + 47 + 58}{5} = 39.2 \text{ sec}$$

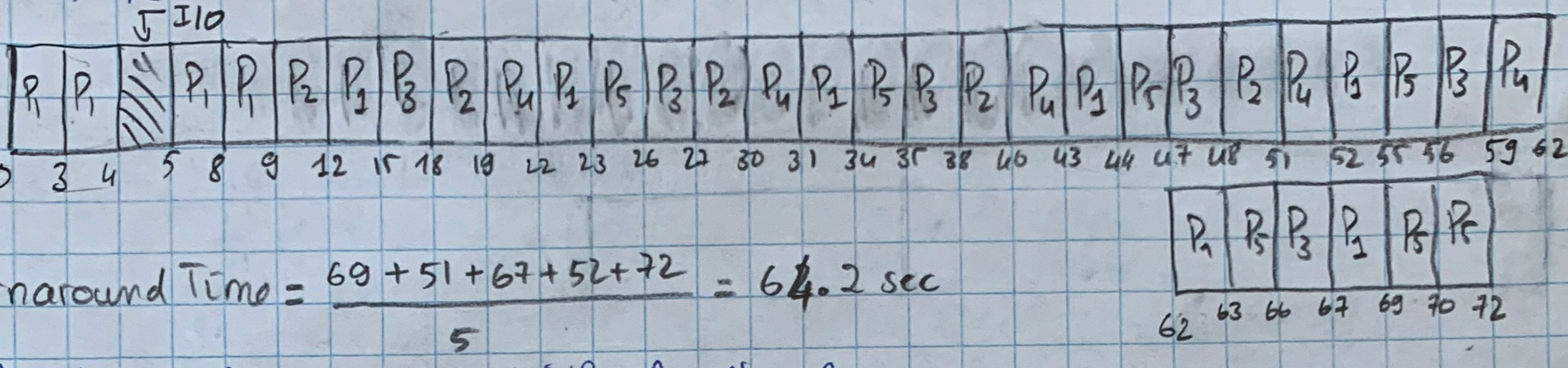
d) Repeat c) with 4 sec I/O:

i) FCSC



Turnaround Time = $\frac{69+48+60+63+71}{5}$ = 62.2 sec

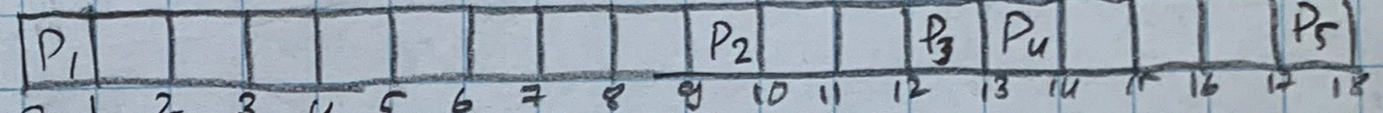
ii) Round Robin (time slice of 3 sec)



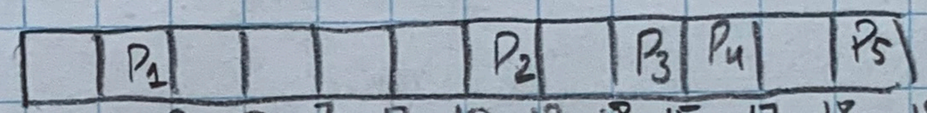
Turnaround Time = $\frac{69+51+67+52+72}{5}$ = 64.2 sec

iii) Multiple queue with feedback

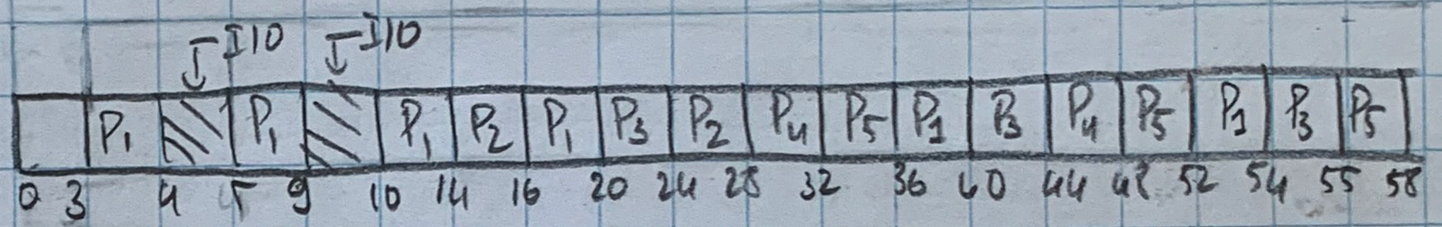
High - Priority Queue:



Mid - Priority Queue:



Low - Priority Queue:



Turnaround Time = $\frac{54+28+44+48+58}{5}$ = 46.4 sec

a) Explain difference

i) FCFS: this scheduling algorithm is the best of the three algorithms that allow to long processes to not get stuck in the queue for a long time. The FCFS completes each process in the order they came in.

ii) RR: This scheduling algorithm is not particulary designed for long processes. In fact, each process has a time slot before the next process takes over; so a long process stays longer in the queue.

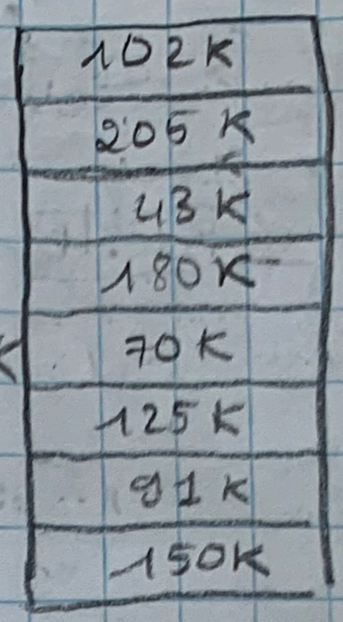
iii) Multilevel feedback queues: this scheduling algorithm works as follows: there is 03 queues and the first two have specific time slots and the last one is a FCFS scheduler. This algorithm tries to eliminate the shortest jobs first through the 2 first queues and the last one is used to eliminate the remaining processes.

b)

i) First fit

- Job 1 will be allocated to the 205K hole [Remaining 83K hole].

New order: 102K, 83K, 43K, 180K, 70K, 125K, 91K, 150K



- Job 2 will be allocated to the 180K hole [Remaining 75K hole].

New order: 102K, 83K, 43K, 75K, 70K, 125K, 91K, 150K

- Job 3 will be allocated to: Null

- Job 4 will be allocated to 102K hole.

New order: 12K, 83K, 43K, 75K, 70K, 125K, 91K, 150K

ii) Best fit

- Job 1 will be allocated to the 125K [3K hole remaining]

New order: 102K, 205K, 43K, 180K, 70K, 3K, 91K, 150K

- Job 2 will be allocated to the 150K hole [45K hole remaining]

New order: 102K, 205K, 43K, 180K, 70K, 3K, 91K, 45K

- Job 3 will be allocated to 205K [Remaining 2K hole]

New order: 102K, 2K, 43K, 180K, 70K, 3K, 91K, 45K

- Job 4 will be allocated to the 91K hole [Remaining 1K hole]

New order: 102K, 2K, 43K, 180K, 70K, 3K, 1K, 45K

iii) Worst fit

- Job 1 will be allocated to the 205K hole [Remaining 83K hole]

New order: 102K, 83K, 43K, 180K, 70K, 125K, 91K, 150K

- Job 2 will be allocated to the 180K hole [Remaining 75K hole]

New order: 102K, 83K, 43K, 75K, 70K, 125K, 91K, 150K

- Job 3 will be allocated to Null
- Job 4 will be allocated to the 150K hole [Remaining 60K]

New order: 102K, 83K, 43K, 75K, 70K, 125K, 91K, 60K