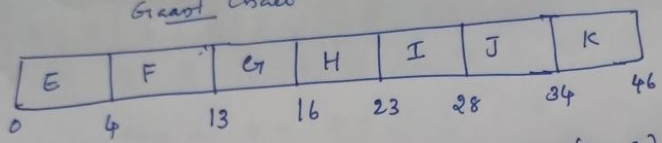
First Come First Serve

Gaant Chart

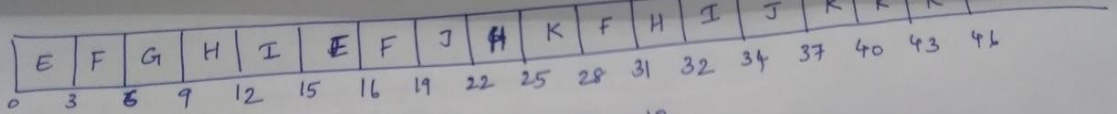


Average waiting time = $\frac{0+(4-2)+(13-3)+(16-5)+(23-11)+(28-17)+(34-26)}{7}$

$= \frac{0+2+10+11+12+11+10}{7} = 8$

Average turnaround time = $\frac{4+13+16+23+28+34+46}{7} = \frac{164}{7} = 23.43$

Round Robin Scheduling (Quantum - 3)



Average waiting time for E = $0+(15-3) = 12$

waiting time for F = $(3-2)+(16-6)+(28-19) = 1+10+9 = 20$

waiting time for G = $(6-3) = 3$

waiting time for H = $(9-5)+(22-12)+(31-25) = 4+10+6 = 20$

waiting time for I = $(12-11)+(32-15) = 1+17 = 18$

waiting time for J = $(19-17)+(34-22) = 2+12 = 14$

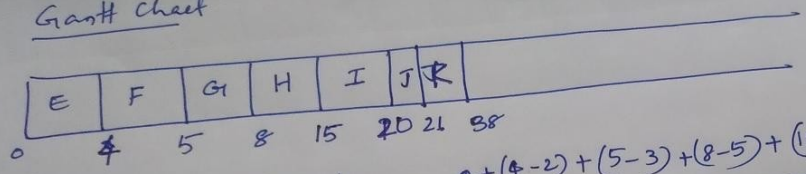
waiting time for K = $(25-26)+(37-28) = 1+9 = 10$

Average waiting time = $\frac{12+20+3+20+18+14+10}{7} = \frac{97}{7} = 13.86$

Average turnaround time = $\frac{16+31+9+32+34+37+46}{7} = \frac{205}{7} = 29.29$

Shortest Remaining Time First

Gantt Chart



Average waiting time = $\frac{0+(4-2)+(5-3)+(8-5)+(15-11)+(20-17)+(26-26)}{7}$

$= \frac{2+2+3+4+3+2}{7} = \frac{16}{7} = 2.28$

Avg Turn around time = $\frac{4+5+8+15+20+26+38}{7} = \frac{116}{7} = 16.57$

Waiting time is shortest for this Algorithm.

2. A.

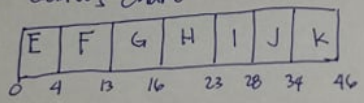
Process	Arrival time	Burst time	Finish time	Turn Around time	Waiting Time
E	0	4	4	4	0
F	2	9	13	11	2
G	3	3	16	13	10
H	5	7	23	18	11
I	11	5	28	17	12
J	17	6	34	17	11
K	24	12	46	22	10

Formula:

Turn Around Time = Finish time − Arrival time

Waiting Time = Turnaround − Burst time

B. Gantt's chart

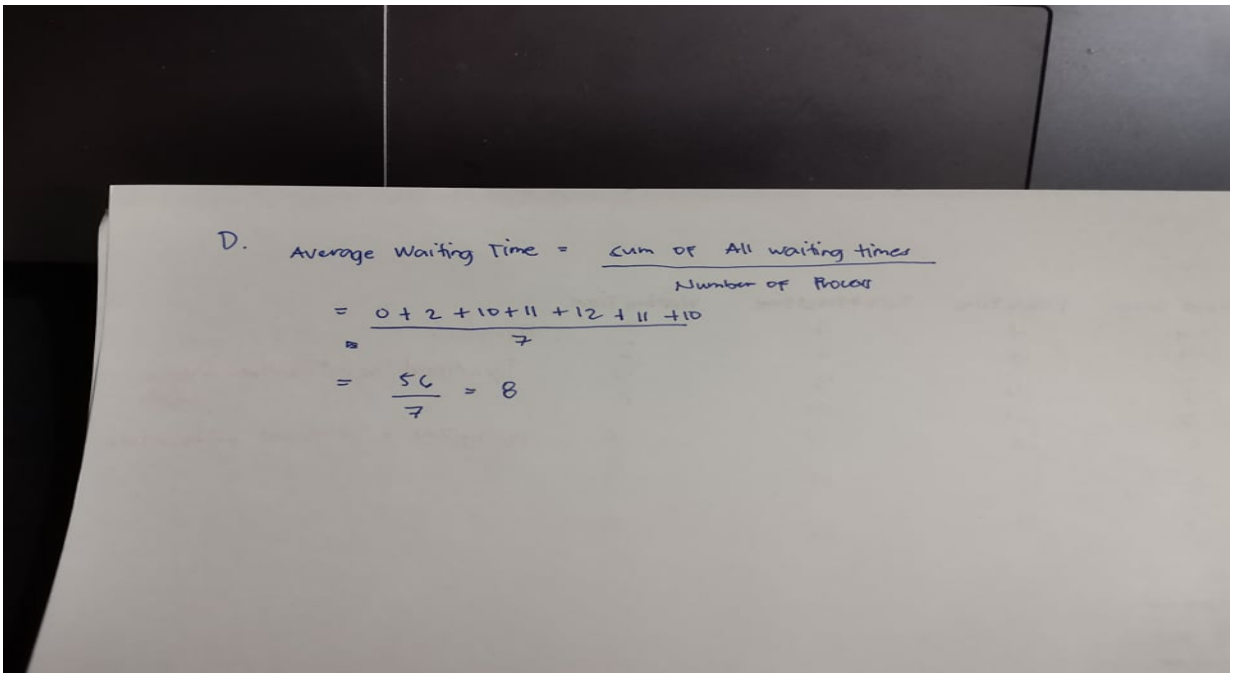


C. Average Turnaround time = $\frac{\text{Sum of All turn Around times}}{\text{Number of Process}}$

$= \frac{4+11+17+18+17+17+22}{7}$

$= \frac{102}{7}$

$= 14.571$



a. Among the three (3) process scheduling algorithms that you have performed, which do you think is the most efficient and why?
   - Sometimes FCFS algorithm is better than the other in short burst time while Round Robin is better for multiple processes in every single time. However, it cannot be predicted what process will come after.

b. Cite significant differences in the results of applying the First-Come First-Serve algorithm and Round Robin algorithm. Elaborate on your answer.
   - FCFS algorithm doesn't include any complex logic, it just puts the process requests in a queue and executes it one by one. Hence, FCFS is pretty simple and easy to implement. Eventually, every process will get a chance to run, so starvation doesn't occur.
   - Round Robin (RR) Each process is served by the CPU for a fixed time quantum, so all processes are given the same priority. Starvation doesn't occur because for each round robin cycle, every process is given a fixed time to execute. No process is left behind.

c. What could possibly happen if the value of the Quantum in Round Robin is increased to 5? Elaborate on your answer.
   - Increasing time quantum for one process may affect completion time of other processes as waiting time of other processes may increase.

d. In your opinion, why does the average turnaround time and waiting time vary per algorithm?
   - Different CPU scheduling algorithms produce different turnaround time for the same set of processes. This is because the waiting time of processes differ when we change the CPU scheduling algorithm.

e. Would you suggest the utilization of the Round Robin algorithm for process scheduling in a file management system? Why or why not?

- Yes, Round-robin scheduling is simple, easy to implement, and starvation-free. Round-robin scheduling can be applied to other scheduling problems, such as data packet scheduling in computer networks.