Project 2: CPU Scheduling Algorithms Analysis

Group: 4
Members: Jesudas Kuruvilla George | Hajira Mohammad Ateequr Rahman | Shuze Liu | Wenhan Kong

1. Introduction

This experiment evaluates six CPU scheduling algorithms: FCFS, SJF, SRT, Round Robin (RR), and Highest Priority First (Preemptive & Non-Preemptive). The simulation enforced a strict start cutoff at time quantum 99. The goal was to compare efficiency, responsiveness, and throughput across 5 random workload runs.

2. Experimental Results

The table below summarizes the aggregate statistics averaged over 5 runs.

Algorithm	Avg TAT	Avg Wait	Avg Resp	Throughput (Jobs/Run)
SRT	**13.97**	**8.74**	7.16	**18.95**
SJF	15.37	10.13	10.13	18.89
FCFS	21.80	16.29	16.29	18.02
HPF-NP	21.88	16.37	16.37	18.02
HPF-Pre	28.91	23.24	8.57	17.49
RR	31.57	25.89	**3.67**	17.48

3. Analysis & Discussion

Efficiency (Turnaround & Wait Time):

Shortest Remaining Time (SRT) was the most efficient algorithm, achieving the lowest Average Turnaround Time (13.97) and Waiting Time (8.74). By preemptively prioritizing jobs closest to completion, SRT clears the queue rapidly. **SJF** followed closely, while **FCFS** performed significantly worse due to the "convoy effect," where short processes wait behind long CPU bursts.

Responsiveness:

Round Robin (RR) demonstrated superior responsiveness with an Average Response Time of **3.67** (approx. 4.4x faster than FCFS). This confirms RR is the ideal choice for interactive systems. However, this responsiveness comes at the cost of the highest Turnaround Time (31.57), as processes remain in the system longer while sharing the CPU.

Throughput & The Cutoff Rule:

SRT and SJF achieved the highest throughput (~19 processes). Because they clear short jobs quickly, the CPU is frequently freed, allowing "late arrival" jobs (time 90-99) to start before the cutoff. Slower algorithms like **RR** often kept the CPU busy with older work past time 99, causing valid new arrivals to be dropped, resulting in lower throughput (~17.5).

Priority Scheduling:

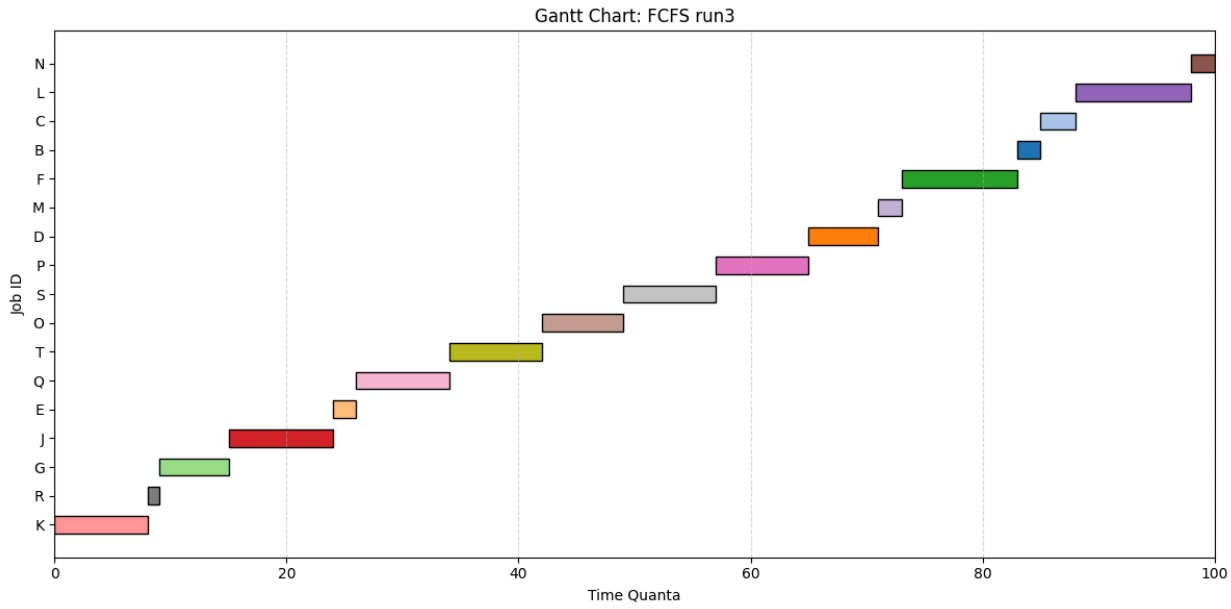
HPF-Preemptive offered better Response Times (8.57) than **HPF-NonPreemptive** (16.37) by allowing high-priority jobs to seize the CPU immediately. However, the constant context switching and re-queueing of lower-priority jobs inflated the overall Turnaround Time for the preemptive version.

4. Conclusion

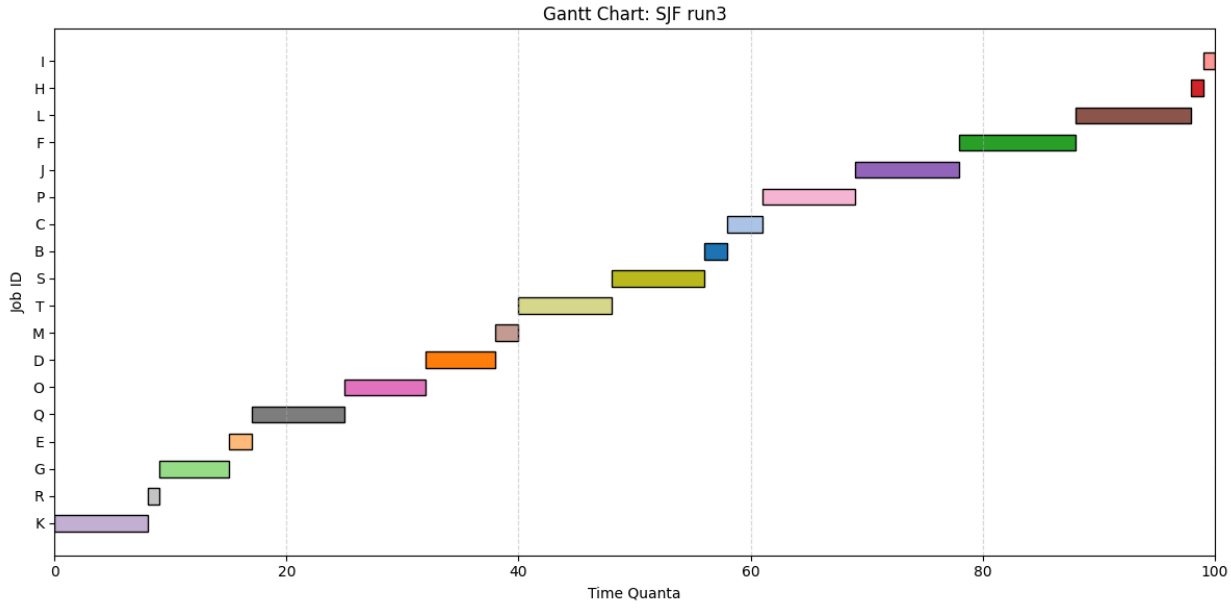
- **SRT** is optimal for batch processing, maximizing throughput and minimizing wait times.
- **Round Robin** is essential for interactive environments; its low response time ensures fairness despite high turnaround times.
- **HPF** successfully stratifies jobs by importance, though it risks starvation for low-priority tasks without aggressive aging.

Appendix Gantt Chart Visualizations

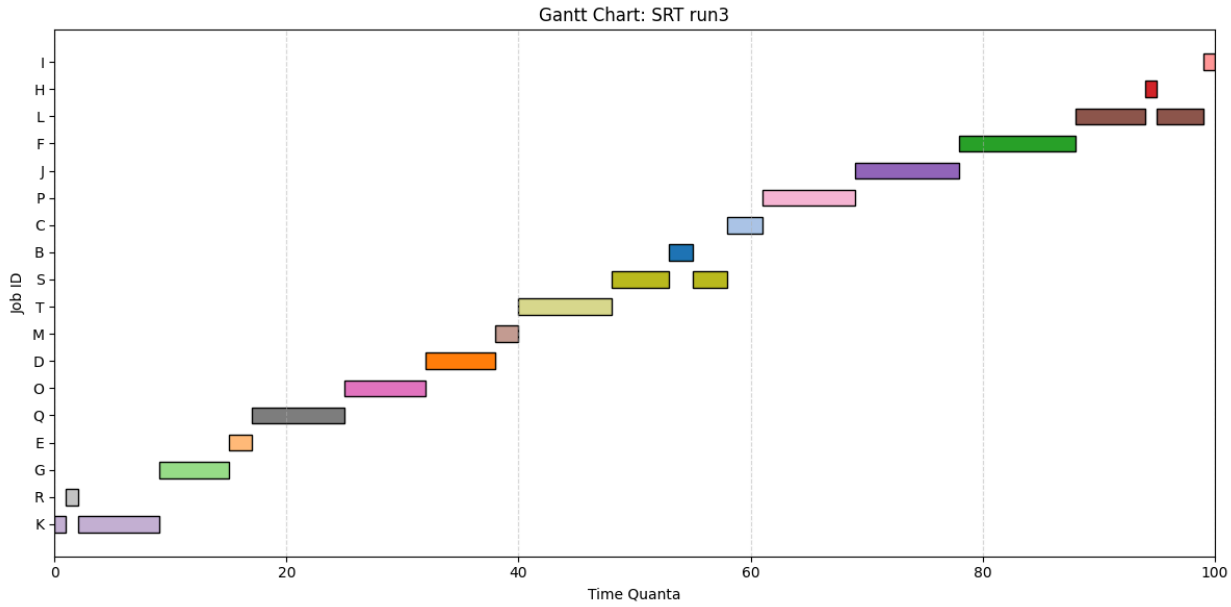
A. First Come First Serve (FCFS)



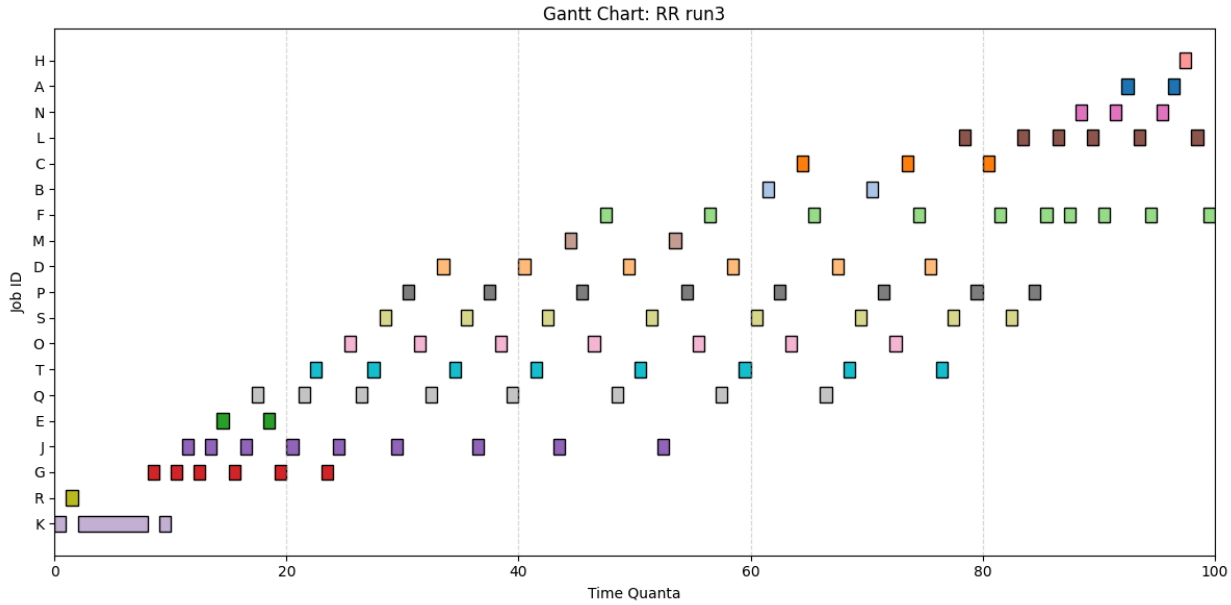
B. Shortest Job First (SJF)



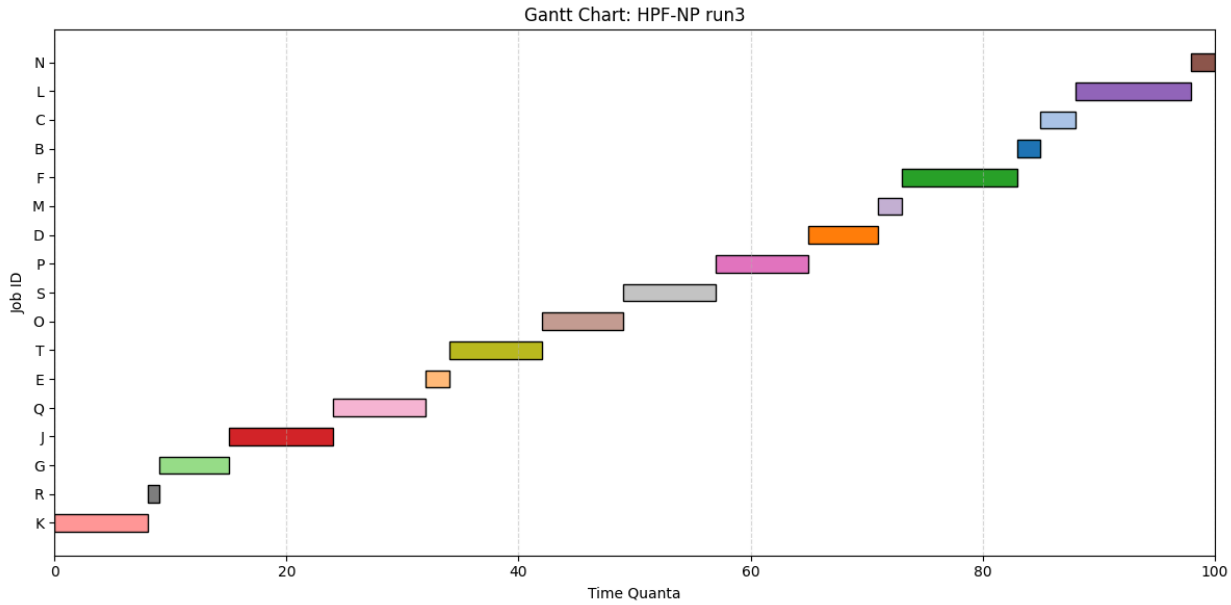
C. Shortest Remaining Time (SRT)



D. Round Robin (RR)



E. HPF Non-Preemptive



F. HPF Preemptive