Assignment 4:
Shortest Job First & Shortest Remaining Time First

Q1) Explanation of Scheduling Schemes:

1. Shortest Job First (SJF) Scheduling:
   - SJF scheduling selects the process with the shortest remaining burst time from the ready queue.
   - It prioritizes shorter jobs to minimize waiting time and improve turnaround time.
   - Processes are executed until completion without preemption.
2. Shortest Remaining Time First (SRTF) Scheduling:
   - SRTF is a preemptive version of SJF, where the running process can be preempted if a new process with a shorter burst time arrives.
   - It continuously selects the process with the shortest remaining burst time, possibly interrupting the currently running process.

Q2) Expected Job Characteristics:

- For **SJF**: Processes with shorter burst times are favored, resulting in reduced waiting time and turnaround time compared to other scheduling algorithms..
- For **SRTF**: Similar to SJF but more suitable for dynamic environments where burst times vary and processes arrive frequently. It excels in scenarios where preemption can significantly improve response time.

Q3) Test Process Data for Suitability:

- For SJF: We achieve optimal performance when jobs arrive in ascending order of CPU burst time. Additionally, it is ideal for interactive operations with small CPU bursts.
  Example:
  0 10 -1
  1 15 -1
  2 20 -1
- For SRTF: Similar to SJF, but with a focus on dynamic scenarios where burst times change frequently and processes arrive continuously. The test should highlight SRTF's effectiveness in preempting longer processes to execute shorter ones and improve response time in such environments.
  The SRTF algorithm outperforms the SJF algorithm by processing jobs faster if context switching time is not considered.

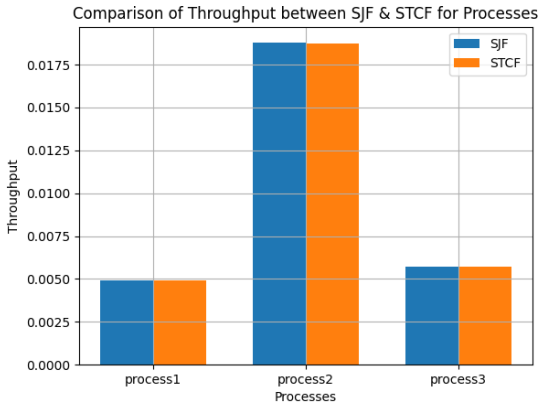
Q4) Test Process Data for Shortcomings:

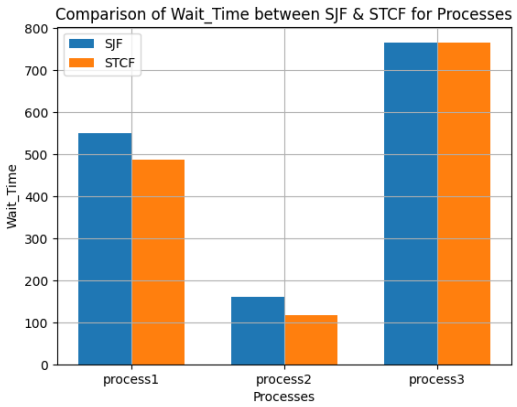
- For SJF: A dataset where long processes arrive first, followed by shorter ones. In this case, SJF may lead to starvation for shorter processes.
  Example: 0 500 -1
  10 100 -1
  25 5 -1
- For SRTF: A dataset where frequent preemptions occur due to the arrival of shorter processes, leading to higher CPU overhead and turnaround time also increases when several large jobs arrive.
  Example: Frequent context switching
  0 5 -1
  1 3 -1
  2 1 -1
  Example: Frequent context switching
  0 100 -1
  0 300 -1
  2 250 -1

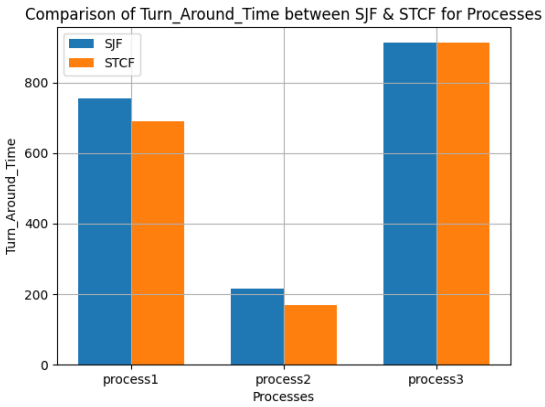
Assumptions: If two or more jobs have shortest remaining time first schedule the job that arrived first.

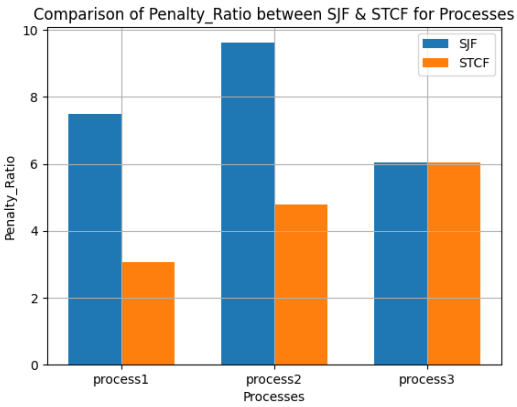
- Metrics Computed from the Scheduling Algorithms:
  - Turnaround time = Completion Time – Arrival Time
  - Waiting time = Turnaround Time – Total CPU burst Time
  - Penalty Ratio = (Turnaround Time-Waiting time) / Turnaround Time
  - Throughput = Total Process/Total Time
  - Response time = Arrival time – Time of First Execution

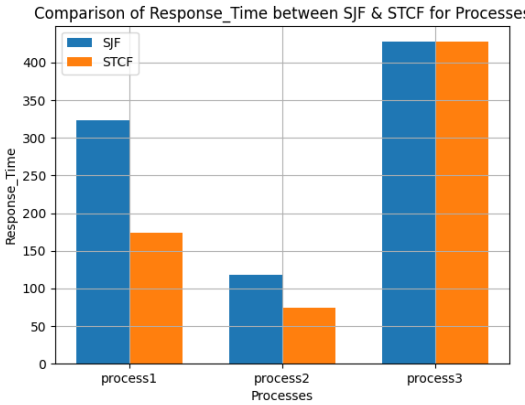
Comparison of SJF and STCF for different Processes:











Observation:

1. Both algorithms give the same throughput time for process1, process2, process3.
2. Wait time for SJF is more than or equal to the wait time of STCF this is because SJF starves the longer running job.
3. Since turnaround time is directly influenced by waiting time, it tends to exhibit a similar pattern as waiting time for both algorithms.
4. The pattern of penalty ratio time is similar to that of turnaround time, as the total burst time remains the same for the processes regardless of the algorithm.
5. Response Time for STCF is usually lower than that of SJF since pre-emption is possible. But in case all the short jobs arrive prior to the longer jobs then CPU will always execute the shorter jobs, this will make the response time same in both cases.