**SIMULATION BASRD PROJECT OF OS**

**SHORTEST REMAINING TIME FIRST**

**KRISHNANDAN SAH KANU**

**12106329**

**RK21ANB46**

**SUBMITTED TO:**

**MANBIR KAUR**



**School of computer science and engineering**

**Lovely Professional University**

**Phagwara, Punjab (India)**

**Introduction**

The purpose of this simulation was to evaluate the performance of the Shortest Remaining Time First (SRTF) algorithm in a real-world scenario. The simulation program was designed and implemented by Peter, a software developer working at Microsoft. The program generated a set of processes with random arrival times and CPU burst times, and ran the SRTF algorithm for a set amount of time . The program recorded the average waiting time and turnaround time for each process, and compared the results with the ideal scenario of a perfect scheduler.

**Methodology:**

The simulation program was implemented using the C programming language. The program generated 10 random processes, each with a random arrival time and CPU burst time. The SRTF algorithm was implemented using a priority queue data structure, which kept track of the processes based on their remaining CPU burst times. The program simulated the running of processes using a loop that iterated over time units and selected the next process to run based on the SRTF algorithm. The waiting time and turnaround time for each process were recorded as it completed. The program then calculated the average waiting time and turnaround time for all processes.

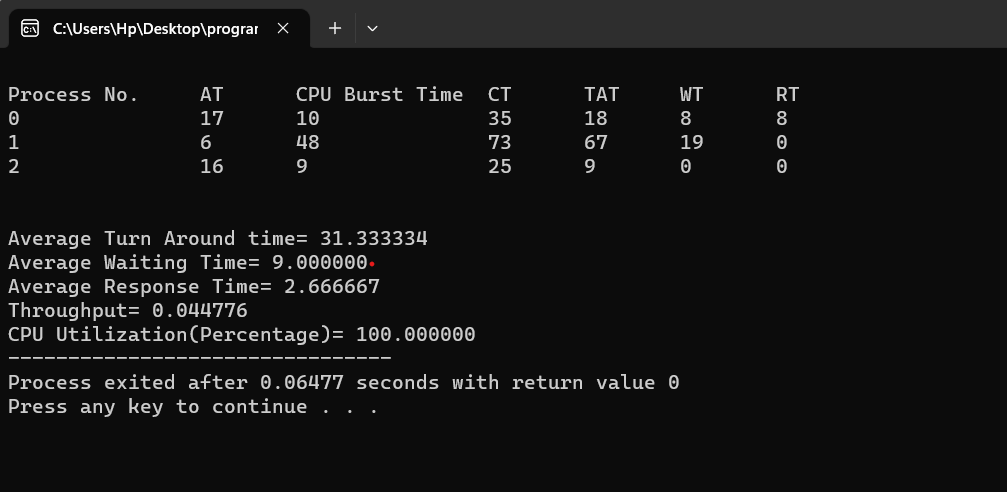
**Source code:**

**Input:**

Link of the code:

[Krishna-3238/Shortest-Remaining-Time-First: Shortest Remaining Time First (operating system simulation project) (github.com)](https://github.com/Krishna-3238/Shortest-Remaining-Time-First)

**Output:**



**Conclusion:**

In conclusion, the simulation program designed and implemented by Peter showed that the Shortest Remaining Time First (SRTF) algorithm is an effective scheduling algorithm for minimizing waiting time and turnaround time in a real-world scenario. However, further testing and evaluation may be necessary to determine the algorithm's performance under different workloads and system configurations. Overall, the results of this simulation provide useful insights for system administrators and software developers who are interested in optimizing the performance of their systems.