Preemptive Scheduler Using Threads

Objective:

The objective of this project is to implement and evaluate various preemptive scheduling algorithms. Specifically, we have implemented the Longest Remaining Time First (LRTF), Shortest Remaining Time First (SRTF), Priority, and Round Robin scheduling algorithms. The goal is to compare their performance in terms of turnaround time and waiting time. Implement the scheduling algorithms to understand their underlying mechanisms and functionality. This includes coding the algorithms in a suitable programming language and designing the necessary algorithms to support process management and scheduling. This report presents the findings of our study, including the methodology, results, and conclusions.

By achieving these objectives, this project aims to enhance our understanding of preemptive scheduling algorithms and their impact on process management in operating systems. The findings and recommendations from this project can assist system administrators and developers in making informed decisions when selecting scheduling algorithms for their systems, thereby improving overall system performance and efficiency.

Introduction:

In the realm of operating systems, scheduling algorithms play a critical role in managing and optimizing the execution of processes or tasks. These algorithms determine how the CPU resources are allocated to

different processes, affecting important performance metrics such as turnaround time, waiting time, and response time. Preemptive scheduling algorithms, in particular, allow for the interruption and resumption of processes based on specific criteria.

Background:

Each scheduling algorithm has its own unique characteristics and prioritization criteria. LRTF and SRTF focus on the remaining execution time of a process, where the former selects the process with the longest remaining time, while the latter selects the process with the shortest remaining time. Priority scheduling assigns a priority value to each process and selects the highest priority process for execution. Round Robin scheduling assigns a fixed time quantum to each process and rotates between processes, allowing each process to execute for a specified time before moving to the next.

Platform and Languages:

For the implementation of the preemptive scheduling algorithms, we utilized a programming platform that supports process management and scheduling. We chose to implement the algorithms in C++, a versatile and widely used programming language. C++ provides a range of libraries (like pthreads.h and semaphore.h) and modules that facilitate the creation and analysis of scheduling algorithms.

Methodology:

- Data Preparation:
  - We asked the user for a set of processes with arrival times, burst times, and priority values to create a diverse workload for testing the algorithms.

- LRTF Algorithm:
  - We implemented the Longest Remaining Time First algorithm, where processes are selected based on the longest remaining burst time.
- SRTF Algorithm:
  - We implemented the Shortest Remaining Time First algorithm, where processes are selected based on the shortest remaining burst time.
- Priority Algorithm:
  - We implemented the Priority algorithm, where processes are selected based on priority values assigned to each process.
- Round Robin Algorithm:
  - We implemented the Round Robin algorithm, where processes are executed in a cyclical manner with a fixed time quantum.
- Performance Evaluation:
  - We measured and compared the turnaround time and waiting time of each algorithm using the generated workload.

Timeline:

The project completed in 2 months, broken down into the following milestones:

Month 1:

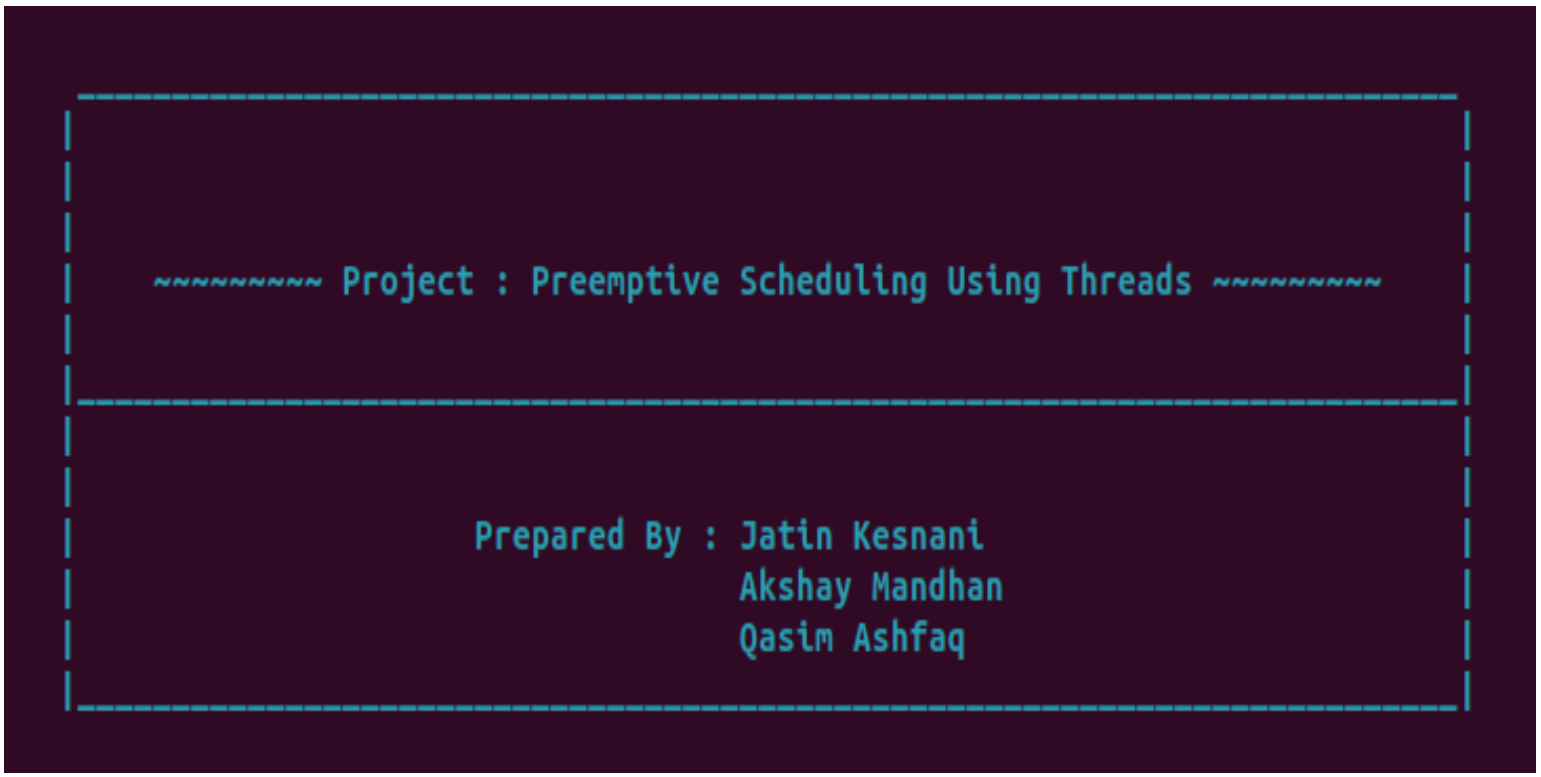
- Week 1: Researching and understanding the concept of preemptive scheduling and how it is implemented using threads.
- Week 2: Identifying the programming language and tools to be used for the project.

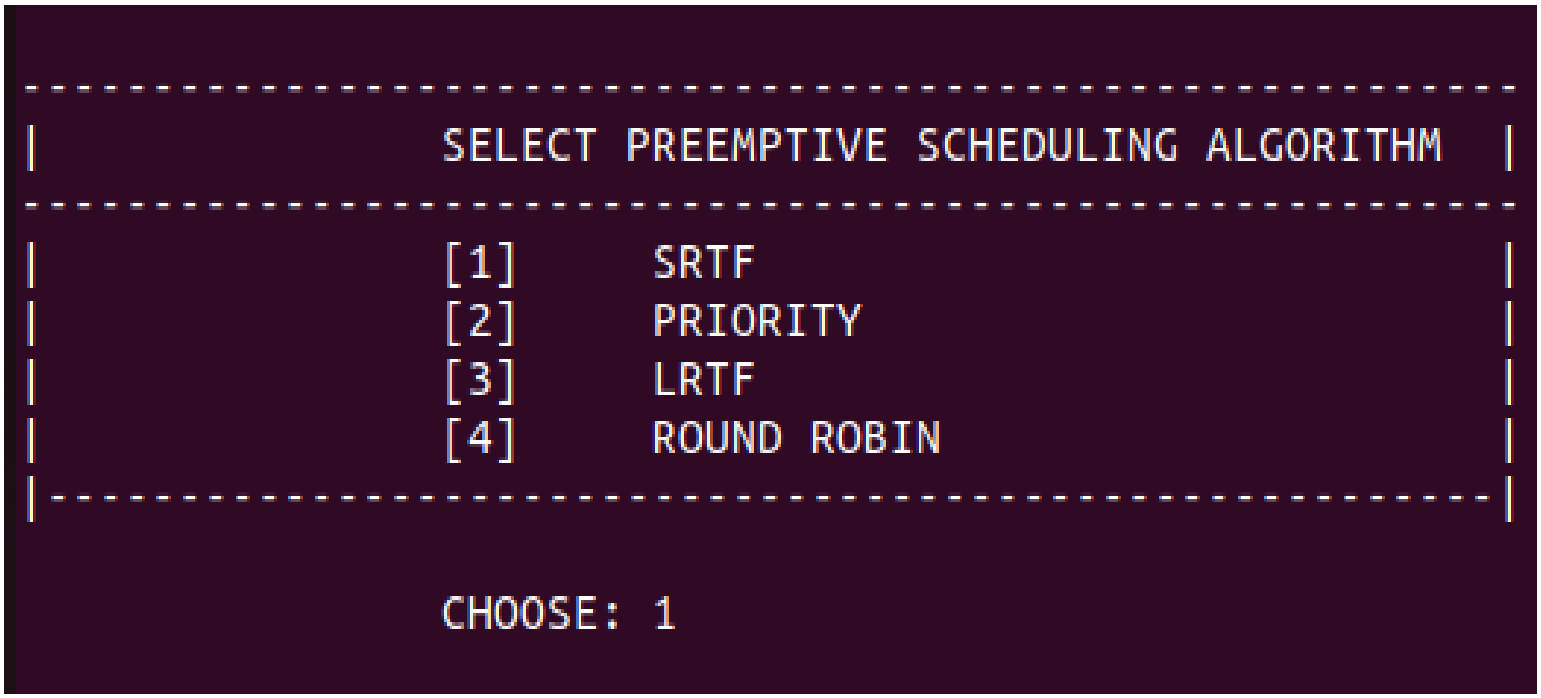
- Week 3: Implementing the basic threading functionality to create and manage threads.
- Week 4: Writing code for the preemptive scheduler to decide which thread should run next and implement thread switching.

Month 2:

- Week 1: Testing the basic functionality of the scheduler and make necessary improvements.
- Week 2: Implementing synchronization mechanisms such as mutexes and semaphores to prevent race conditions.
- Week 3: Testing the entire system and debugging any issues found.
- Week 4: Writing documentation and user manual to explain how to use the scheduler.

Project Gallery:





Shortest Remaining Time First

```
-----------------------------
| Process | Arrival | Burst |
|   ID    |  Time   | Time  |
-----------------------------
|   P1    |    3    |   6   |
|   P2    |    6    |   5   |
|   P3    |    4    |   4   |
|   P4    |    3    |   2   |
|   P5    |   10    |   5   |
|   P6    |    8    |   8   |
|   P7    |    7    |   4   |
|   P8    |   11    |   3   |
|   P9    |    9    |   2   |
|   P10   |    2    |   7   |
-----------------------------

Process: P4 has arrived! thread created for P4
Process: P2 has arrived! thread created for P2
Process: P3 has arrived! thread created for P3
Process: P6 has arrived! thread created for P6
Process: P1 has arrived! thread created for P1
Process: P7 has arrived! thread created for P7
Process: P8 has arrived! thread created for P8
Process: P5 has arrived! thread created for P5
Process: P9 has arrived! thread created for P9
Process: P10 has arrived! thread created for P10

-------------------------------------------------------------------
| Process | Arrival | Burst | Finishing | Turnaround | Waiting |
|   ID    |  Time   | Time  |   Time    |    Time    |  Time   |
-------------------------------------------------------------------
|   P1    |    3    |   6   |    40     |     37     |   31    |
|   P2    |    6    |   5   |    23     |     17     |   12    |
|   P3    |    4    |   4   |     9     |      5     |    1    |
|   P4    |    3    |   2   |     5     |      2     |    0    |
|   P5    |   10    |   5   |    28     |     18     |   13    |
|   P6    |    8    |   8   |    48     |     40     |   32    |
|   P7    |    7    |   4   |    18     |     11     |    7    |
|   P8    |   11    |   3   |    14     |      3     |    0    |
|   P9    |    9    |   2   |    11     |      2     |    0    |
|   P10   |    2    |   7   |    34     |     32     |   25    |
-------------------------------------------------------------------

Average Waiting Time    : 12.1
Average Turnaround Time : 16.7

                --> GANTT Chart <--
 ---------------------------------------------------------------
 | ~~ | P10 | P4 | P3 | P9 | P8 | P7 | P2 | P5 | P1 | P10 | P6 |
 ---------------------------------------------------------------
 0    2     3    5    9   11   14   18   23   28   34    40   48
```

Longest Remaining Time First

```
----------------------------
| Process | Arrival | Burst |
|   ID    |  Time   | Time  |
----------------------------
|   P1    |    3    |   6   |
|   P2    |    6    |   5   |
|   P3    |    4    |   4   |
|   P4    |    3    |   2   |
|   P5    |   10    |   5   |
|   P6    |    8    |   8   |
|   P7    |    7    |   4   |
|   P8    |   11    |   3   |
|   P9    |    9    |   2   |
|   P10   |    2    |   7   |
----------------------------

Process: P1 has arrived! thread created for P1
Process: P5 has arrived! thread created for P5
Process: P6 has arrived! thread created for P6
Process: P7 has arrived! thread created for P7
Process: P8 has arrived! thread created for P8
Process: P9 has arrived! thread created for P9
Process: P10 has arrived! thread created for P10
Process: P3 has arrived! thread created for P3
Process: P4 has arrived! thread created for P4
Process: P2 has arrived! thread created for P2

-------------------------------------------------------------
| Process | Arrival | Burst | Finishing | Turnaround | Waiting |
|   ID    |  Time   | Time  |   Time    |    Time    |  Time   |
-------------------------------------------------------------
|   P1    |    3    |   6   |    41     |     38     |   32    |
|   P2    |    6    |   5   |    43     |     37     |   32    |
|   P3    |    4    |   4   |    42     |     38     |   34    |
|   P4    |    3    |   2   |    40     |     37     |   35    |
|   P5    |   10    |   5   |    47     |     37     |   32    |
|   P6    |    8    |   8   |    45     |     37     |   29    |
|   P7    |    7    |   4   |    44     |     37     |   33    |
|   P8    |   11    |   3   |    48     |     37     |   34    |
|   P9    |    9    |   2   |    46     |     37     |   35    |
|   P10   |    2    |   7   |    39     |     37     |   30    |
-------------------------------------------------------------

Average Waiting Time    : 32.6
Average Turnaround Time : 37.2

               --> GANTT Chart <--
 ---------------------------------------------------------------------------------------------------------------------------------------------------------------
 | ~~ | P10 | P1 | P2 | P10 | P6 | P5 | P1 | P2 | P3 | P6 | P7 | P10 | P1 | P2 | P3 | P5 | P6 | P7 | P8 | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P9 | P10 | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 |
 ---------------------------------------------------------------------------------------------------------------------------------------------------------------
 0   2    4   6   7    8   12  14  15  16  17  18  19   21  22  23  24  25  26  27  29  30  31  32  33  34  35  36  37   39  40  41  42  43  44  45  46  47  48
```

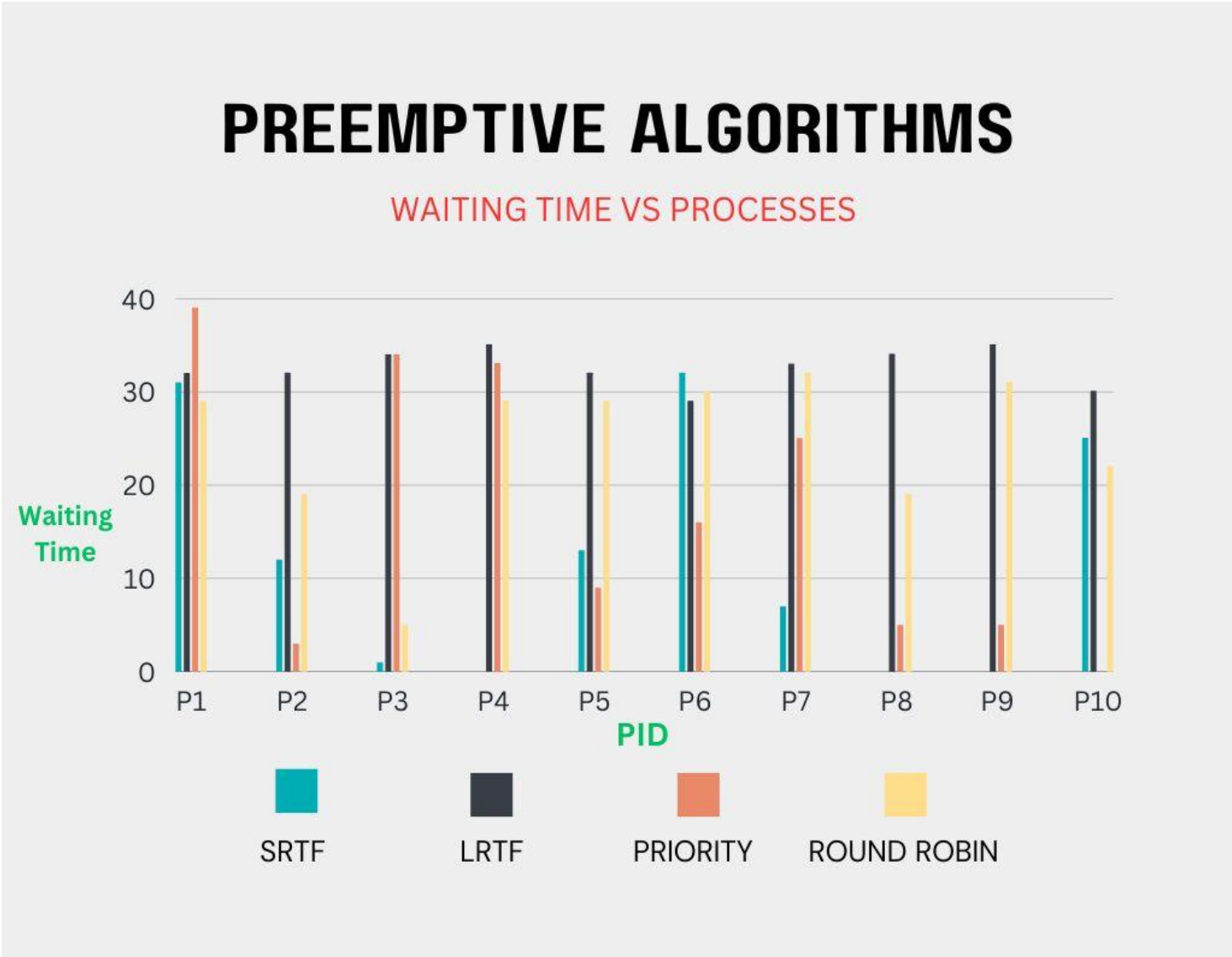
Priority

```
---------------------------------------
| Process | Arrival | Burst | Process  |
|   ID    |  Time   | Time  | Priority |
---------------------------------------
|   P1    |    3    |   6   |     9    |
|   P2    |    6    |   5   |     1    |
|   P3    |    4    |   4   |     8    |
|   P4    |    3    |   2   |     7    |
|   P5    |   10    |   5   |     4    |
|   P6    |    8    |   8   |     5    |
|   P7    |    7    |   4   |     6    |
|   P8    |   11    |   3   |     3    |
|   P9    |    9    |   2   |     2    |
|   P10   |    2    |   7   |     0    |
---------------------------------------

Process: P1 has arrived! thread created for P1
Process: P5 has arrived! thread created for P5
Process: P6 has arrived! thread created for P6
Process: P7 has arrived! thread created for P7
Process: P8 has arrived! thread created for P8
Process: P9 has arrived! thread created for P9
Process: P10 has arrived! thread created for P10
Process: P3 has arrived! thread created for P3
Process: P4 has arrived! thread created for P4
Process: P2 has arrived! thread created for P2

-------------------------------------------------------------------------------
| Process | Arrival | Burst | Process  | Finishing | Turnaround | Waiting |
|   ID    |  Time   | Time  | Priority |   Time    |    Time    |  Time   |
-------------------------------------------------------------------------------
|   P1    |    3    |   6   |     9    |    48     |     45     |   39    |
|   P2    |    6    |   5   |     1    |    14     |      8     |    3    |
|   P3    |    4    |   4   |     8    |    42     |     38     |   34    |
|   P4    |    3    |   2   |     7    |    38     |     35     |   33    |
|   P5    |   10    |   5   |     4    |    24     |     14     |    9    |
|   P6    |    8    |   8   |     5    |    32     |     24     |   16    |
|   P7    |    7    |   4   |     6    |    36     |     29     |   25    |
|   P8    |   11    |   3   |     3    |    19     |      8     |    5    |
|   P9    |    9    |   2   |     2    |    16     |      7     |    5    |
|   P10   |    2    |   7   |     0    |     9     |      7     |    0    |
-------------------------------------------------------------------------------

Average Waiting Time    : 16.9
Average Turnaround Time : 21.5

                --> GANTT Chart <--
 ---------------------------------------------------
 | ~~ | P10 | P2 | P9 | P8 | P5 | P6 | P7 | P4 | P3 | P1 |
 ---------------------------------------------------
 0    2     9   14   16   19   24   32   36   38   42   48
```

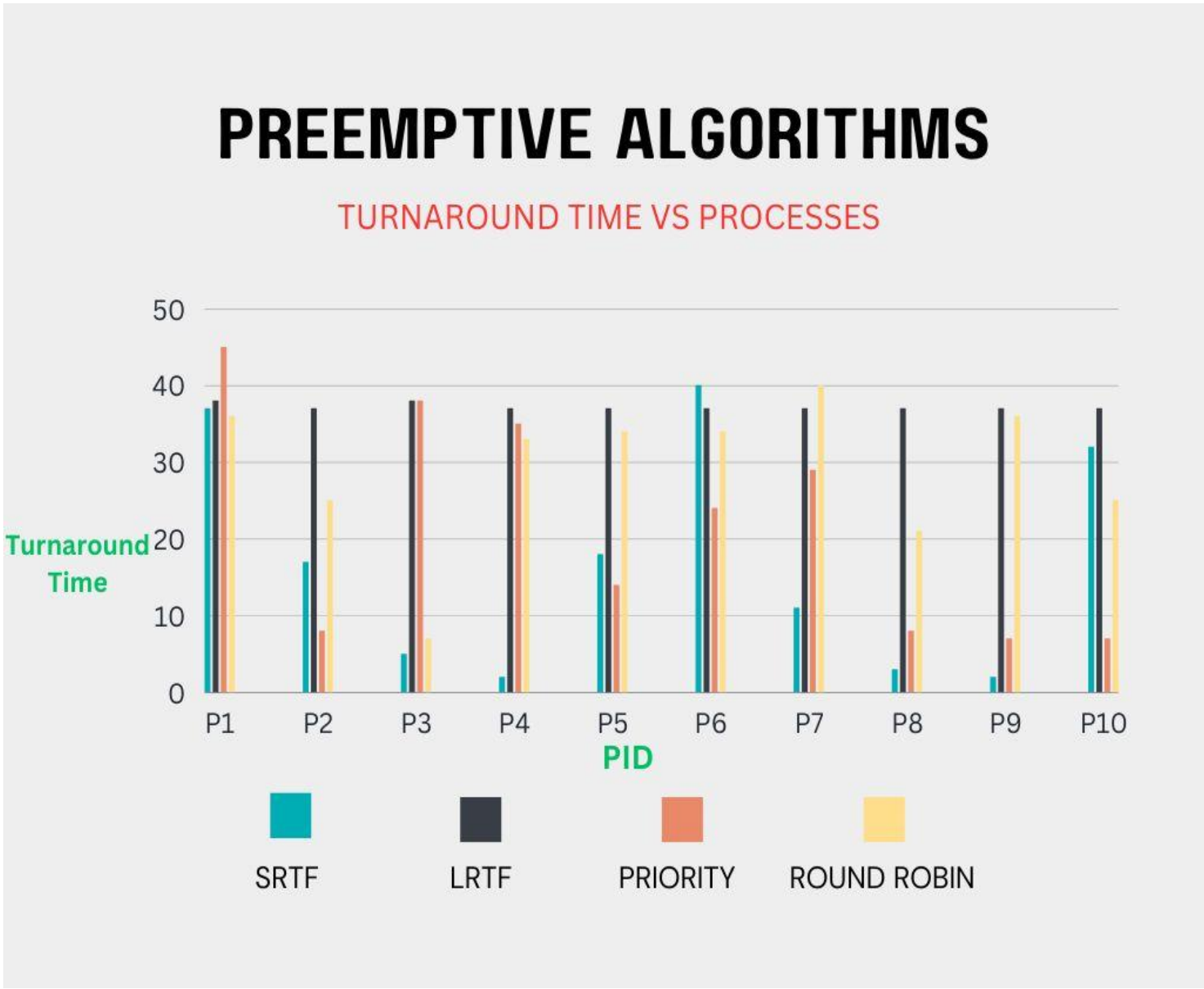
Round Robin

```
---------------------------
| Process | Arrival | Burst |
|   ID    |  Time   | Time  |
---------------------------
|   P1    |    3    |   6   |
|   P2    |    6    |   5   |
|   P3    |    4    |   4   |
|   P4    |    3    |   2   |
|   P5    |   10    |   5   |
|   P6    |    8    |   8   |
|   P7    |    7    |   4   |
|   P8    |   11    |   3   |
|   P9    |    9    |   2   |
|   P10   |    2    |   7   |
---------------------------
Process: P10 has arrived! thread created.
Process: P1 has arrived! thread created.
Process: P4 has arrived! thread created.
Process: P3 has arrived! thread created.
Process: P2 has arrived! thread created.
Process: P7 has arrived! thread created.
Process: P6 has arrived! thread created.
Process: P5 has arrived! thread created.
Process: P8 has arrived! thread created.
Process: P9 has arrived! thread created.

-------------------------------------------------------------
| Process | Arrival | Burst | Finishing | Turnaround | Waiting |
|   ID    |  Time   | Time  |   Time    |    Time    |  Time   |
-------------------------------------------------------------
|   P10   |    2    |   7   |    38     |     36     |   29    |
|   P1    |    3    |   6   |    28     |     25     |   19    |
|   P4    |    3    |   2   |    10     |      7     |    5    |
|   P3    |    4    |   4   |    37     |     33     |   29    |
|   P2    |    6    |   5   |    40     |     34     |   29    |
|   P7    |    7    |   4   |    41     |     34     |   30    |
|   P6    |    8    |   8   |    48     |     40     |   32    |
|   P9    |    9    |   2   |    30     |     21     |   19    |
|   P5    |   10    |   5   |    46     |     36     |   31    |
|   P8    |   11    |   3   |    36     |     25     |   22    |
-------------------------------------------------------------

Average Waiting Time    : 24.5
Average Turnaround Time : 29.1

               --> GANTT Chart <--
 ---------------------------------------------------------------------------------------------------
 | ~~ | P10 | P1 | P4 | P3 | P10 | P2 | P7 | P6 | P1 | P9 | P5 | P8 | P3 | P10 | P2 | P7 | P6 | P5 | P6 |
 ---------------------------------------------------------------------------------------------------
 0    2     5    8    10   13    16   19   22   25   28   30   33   36   37    38   40   41   44   46   48
```

Results:


PREEMPTIVE ALGORITHMS
WAITING TIME VS PROCESSES
Waiting Time
40
30
20
10
0
P1
P2
P3
P4
P5
P6
P7
P8
P9
P10
PID
SRTF
LRTF
PRIORITY
ROUND ROBIN

PREEMPTIVE ALGORITHMS



Conclusion:

Based on the results obtained from our study, we can draw the following conclusions:

- The LRTF algorithm performs well in scenarios where long-running processes need to be prioritized.
- The SRTF algorithm is effective in reducing response time and overall execution time.
- Priority scheduling provides a flexible approach, allowing processes to be executed according to their assigned priority.
- Round Robin scheduling ensures fair execution by providing equal time slices to each process.

The choice of preemptive scheduling algorithm depends on the specific requirements of the system and the workload characteristics. Factors such as responsiveness, fairness, and throughput should be considered when selecting an appropriate algorithm for a given scenario.