



OS COURSE BASED PROJECT

PRIORITY QUEUE VISUALIZATION

OUR TEAM

P.Aashritha Raj

22071A0551

S. Vijaya Charishma

22071A0554

Vellanki Chenchu Dedeepya

22071A0563

V.Sree Charan

23075A0507



ABSTRACT

- Priority queues play a pivotal role in operating systems, particularly in managing tasks with varying levels of urgency. To enhance the understanding and teaching of these data structures, our project focuses on developing a comprehensive visualization tool for priority queue algorithms. This tool graphically demonstrates the core operations of priority queues—such as insertion, deletion, and reordering—providing real-time visual feedback on the queue's state. By leveraging visual aids to represent heap structures, binary trees, and other related data structures, the tool makes abstract concepts more tangible and accessible, thereby bridging the gap between theoretical learning and practical application.

INTRODUCTION

- Priority queues are fundamental components in the field of computer science, particularly within operating systems. They are specialized data structures that manage elements based on their priority levels, ensuring that higher priority tasks are processed before lower priority ones.
- This mechanism is crucial for various operating systems functionalities, such as task scheduling, resource management, and process synchronization. Understanding how priority queues operate and their impact on system performance is essential for anyone studying or working in the domain of operating systems.

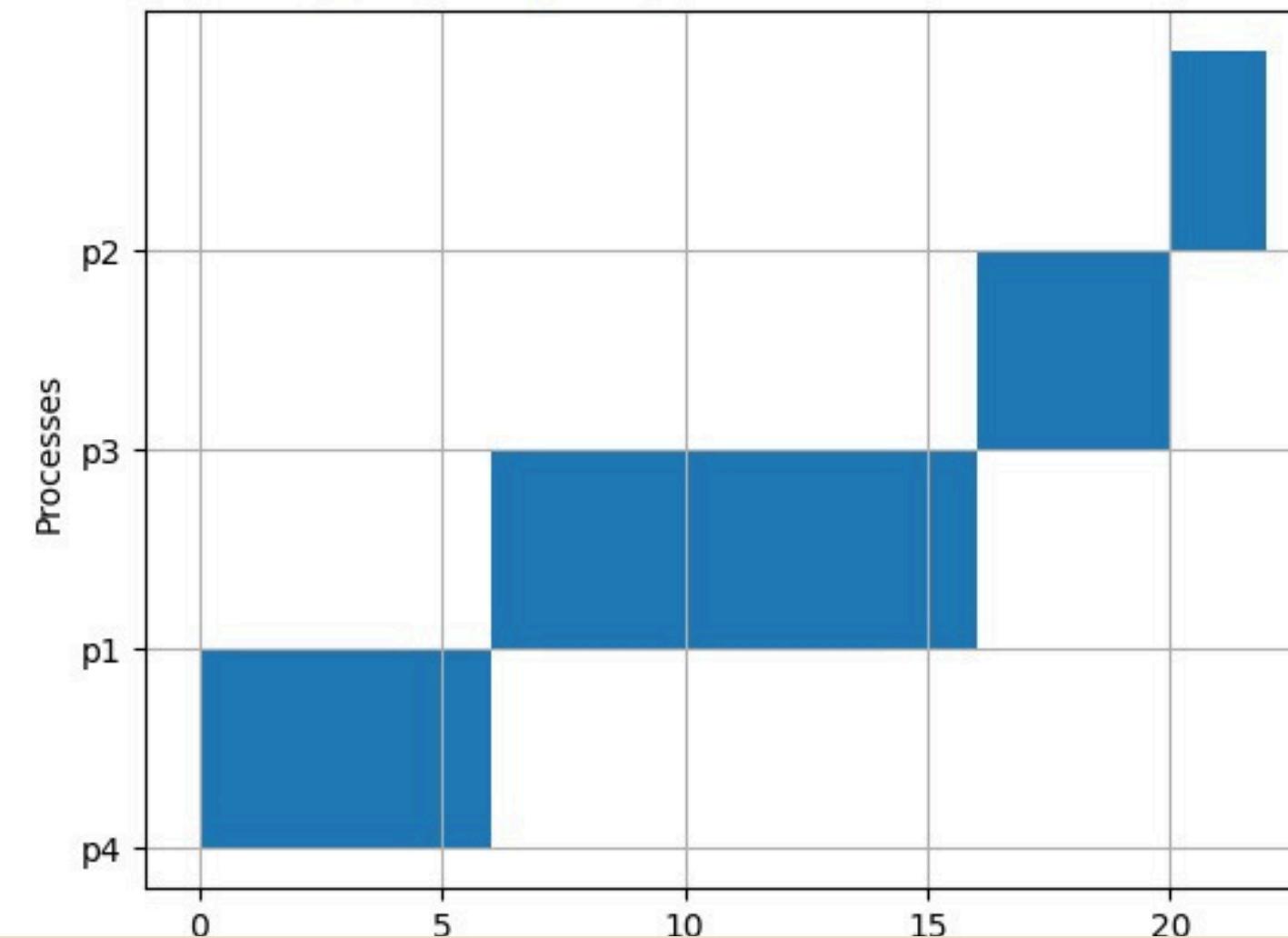
PROBLEM STATEMENT

- Understanding the operations and impact of priority queues in operating systems is a critical yet complex aspect of computer science education. Traditional methods of teaching these concepts often rely heavily on theoretical explanations and static diagrams, which can be difficult for students to fully comprehend. This lack of intuitive understanding hampers the ability of students and professionals to grasp the intricacies of task scheduling, resource management, and process prioritization, which are vital for efficient operating system design and implementation.

OUTPUT

```
enter Burst time for Processes:  
p1: 10  
p2: 2  
p3: 4  
p4: 6  
Enter priority of the processes:  
p1: 2  
p2: 4  
p3: 3  
p4: 1  
Process Burst Time Priority Arrival Time Waiting Time Turnaround Time Completion Time  
p1 10 2 0 6 16 16  
p2 2 4 0 20 22 22  
p3 4 3 0 16 20 20  
p4 6 1 0 0 6 6
```

Execution Order: p4 → p1 → p3 → p2



THANK YOU