Title: Implementation of FCFS Algorithm Using C/C++

Objective: The objective of this lab is to implement the First-Come, First-Served (FCFS) algorithm for process scheduling in an operating system using the C programming language without using structs, objects, classes, or functions.

Theory:

FCFS Algorithm Theoretical Explanation

The First-Come, First-Served (FCFS) algorithm is a straightforward, non-preemptive scheduling technique used in operating systems. It executes processes in the order they arrive in the ready queue, a data structure holding processes awaiting CPU execution. Once a process starts running, it continues until completion or voluntary termination, without interruption.

Here's how the FCFS algorithm works:

1. Initially, the ready queue is empty.

2. When a process arrives, it is added to the tail of the ready queue.

3. The CPU scheduler picks the process at the head of the ready queue for execution.

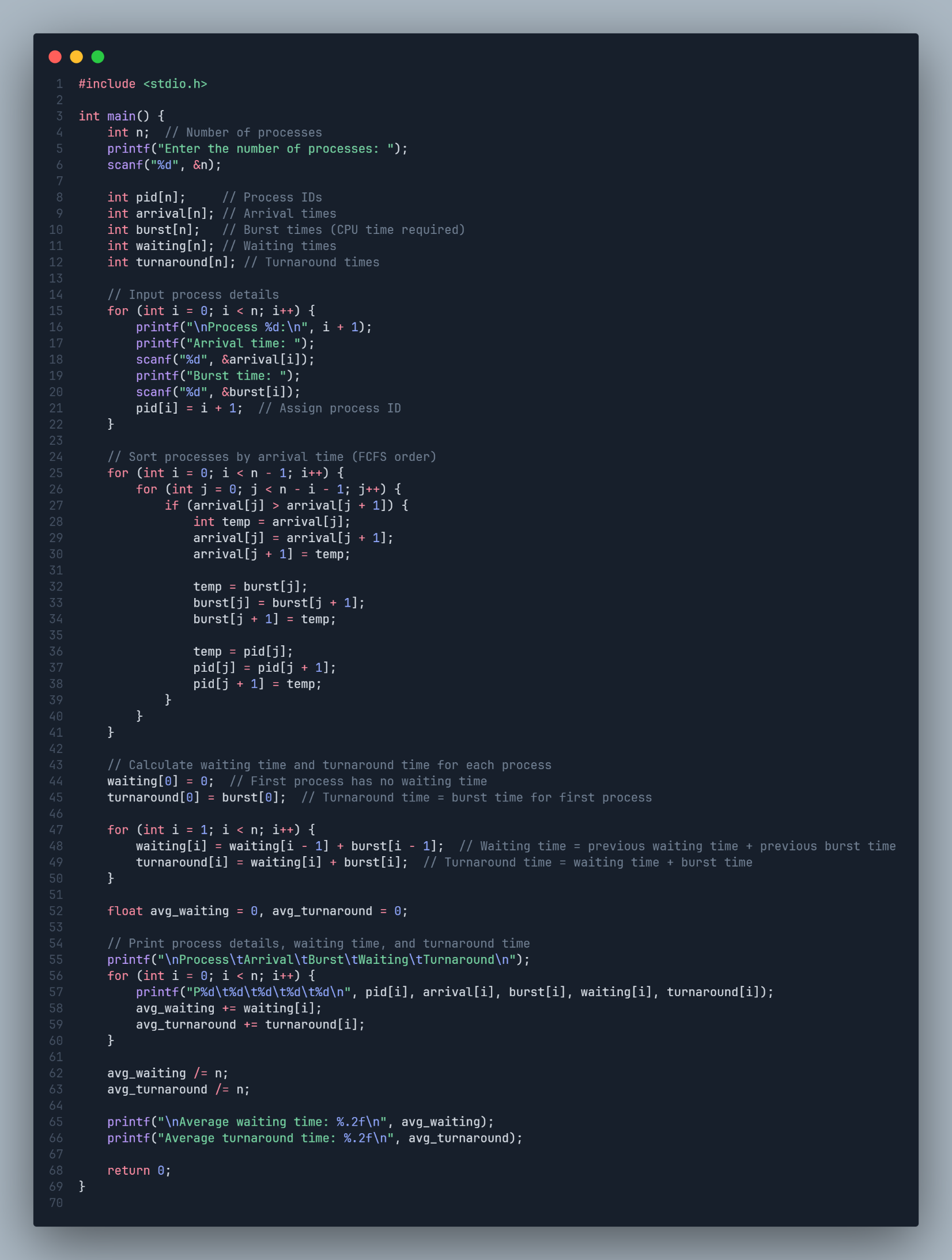
4. The selected process continues to run until it completes or blocks (e.g., due to an I/O operation).

5. If the process completes, it is removed from the ready queue, and the next process in the queue is selected for execution.

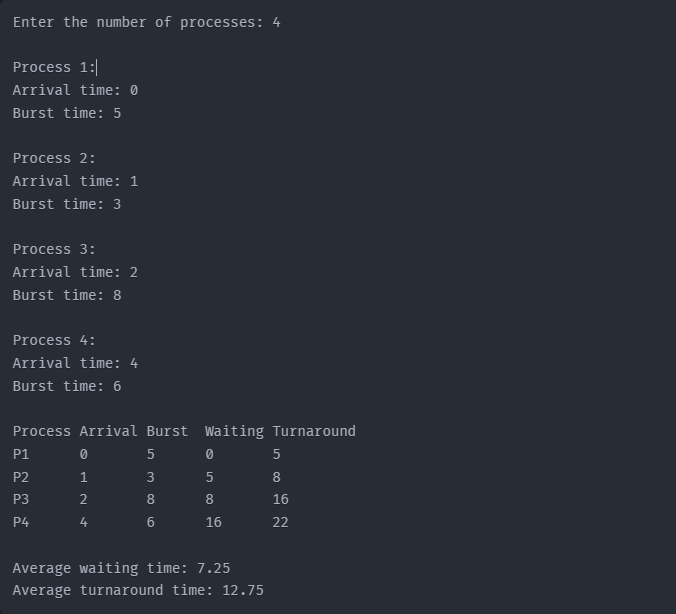
6. If the process blocks, it is removed from the ready queue and placed in a blocked queue until it becomes ready again (e.g., after the I/O operation completes).

7. Once a blocked process becomes ready, it is added to the tail of the ready queue, and the cycle continues.

FCFS Code in C programming:



Sample Output:



Conclusion: The FCFS algorithm is a simple and fair scheduling algorithm, but it can lead to poor performance in certain scenarios, such as when processes with long burst times arrive early and block other processes from executing. More advanced scheduling algorithms, like Shortest Job First (SJF) or Round-Robin (RR), are often used in practice to improve system performance and responsiveness.