**Practical 4**

**Aim**: demonstrate the First Come First Serve CPU.

**Theory:**

FCFS stands for First Come First Serve. In the FCFS scheduling algorithm, the job that arrived first in the ready queue is allocated to the CPU and then the job that came second, and so on. We can say that the ready queue acts as a FIFO (First In First Out) queue thus the arriving jobs/processes are placed at the end of the queue.

FCFS is a non-preemptive scheduling algorithm as a process holds the CPU until it either terminates or performs I/O. Thus, if a longer job has been assigned to the CPU then many shorter jobs after it will have to wait. This algorithm is used in most batch operating systems.

**Characteristics:**

It follows the non-preemptive approach i.e. once a process has control over the CPU it will not preempt until it terminates.

The criteria for the selection of processes is arrival time. The dispatcher selects the first job in the ready queue and this job runs to completion of its CPU burst.

The average waiting time is very high so not optimal and thus gives poor performance.

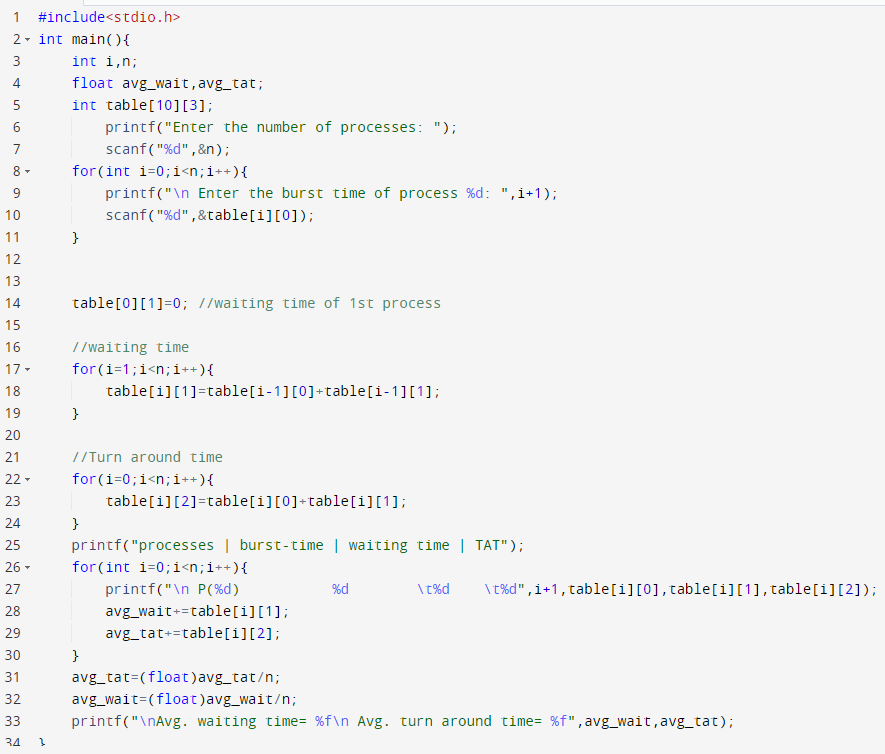
**Advantages:**

* FCFS algorithm is simple, easy to implement into any preexisting system, and easy to understand.
* Better for processes with large burst time as there is no context switch involved between processes.
* It is a fair algorithm as priority is not involved, processes that arrive first get served first.

**Disadvantages:**

* Convoy effect occurs i.e. all small processes have to wait for one large process to get off the CPU.
* It is non-preemptive, the process will not release the CPU until it finishes its task and terminates.
* It is not appropriate for interactive systems as it cannot guarantee a short response time.
* Average waiting time is high and the turnaround time is unpredictable which leads to poor performance.

Code:



Output:

