Experiment No :07

Experiment Name: Implementation of FCFS Scheduling Algorithm.
Objectives:

- What is FCFS Scheduling Algorithm?
- How to implementation in C?

❖ **FCFS Scheduling Algorithm :**

The First Come First Served (FCFS) Scheduling Algorithm is the simplest one. In this algorithm the set of ready processes is managed as FIFO (first-in-first-out) Queue. The processes are serviced by the CPU until completion in order of their entering in the FIFO queue.

A process once allocated the CPU keeps it until releasing the CPU either by terminating or requesting I/O. For example, interrupted process is allowed to run after interrupt handling is done with.

FIRST COME FIRST SERVE SCHEDULING (FCFS)

Description:

To calculate the average waiting time using the FCFS algorithm first the waiting time of the first process is kept zero and the waiting time of the second process is the burst time of the first process and the waiting time of the third process is the sum of the burst times of the first and the second process and so on. After calculating all the waiting times the average waiting time is calculated as the average of all the waiting times. FCFS mainly says first come first serve the algorithm which came first will be served first.

Algorithm:

Step 1: Start the process
Step 2: Accept the number of processes in the ready Queue
Step 3: For each process in the ready Q, assign the process name and the burst time.
Step 4: Set the waiting of the first process as _0'and its burst time as its turnaround time.

Step 5: for each process in the Ready Q calculate

a) Waiting time (n) = waiting time (n-1) + Burst time (n-1)

b) Turnaround time (n)= waiting time(n)+Burst time(n)

Step 6: Calculate

a) Average waiting time = Total waiting Time / Number of process

b) Average Turnaround time = Total Turnaround Time / Number of process

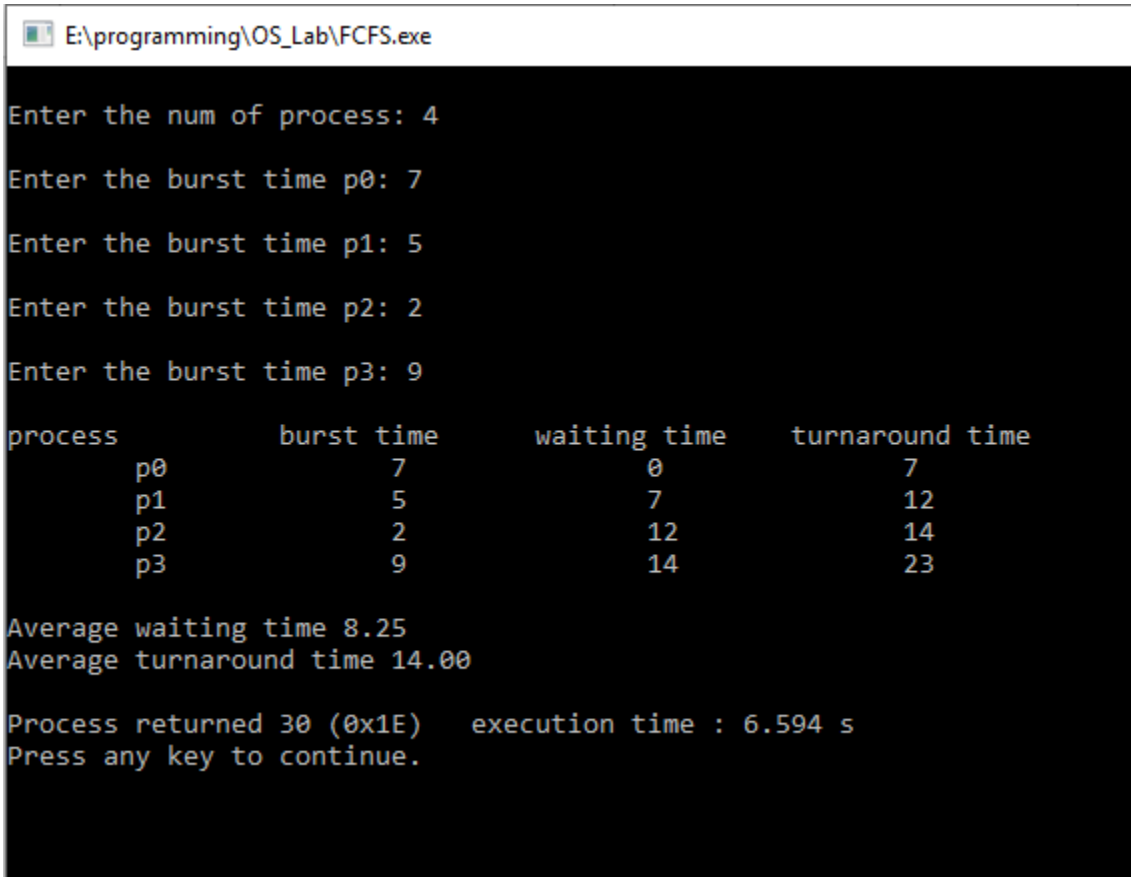
Step 7: Stop the process

Program: To write the c program to implement scheduling algorithm for first come first serve scheduling (FCFS).

```
#include<stdio.h>
#include<conio.h>
void main()
{
   int  n,i,j,bt[30],wt[30],tat[30];
   float awt=0,atat=0;
   printf("\nEnter the num of process: ");
   scanf("%d",&n);
   for(i=0;i<n;i++)
   {
      printf("\nEnter the burst time p%d: ",i);
      scanf("%d",&bt[i]);
   }

   printf("\nprocess \t burst time \t waiting time \t turnaround time\n");
   for(i=0;i<n;i++)
   {
      wt[i]=0;
      tat[i]=0;
      for(j=0;j<i;j++)
```

```
        {
           wt[i]=wt[i]+bt[j];
        }
        tat[i]=wt[i]+bt[i];

        awt=awt+wt[i];
        atat= atat+tat[i];
        printf("\tp%d\t\t%d\t\t%d\t\t%d\n",i,bt[i],wt[i],tat[i]);
    }
    awt=awt/n;
    atat=atat/n;
    printf("\nAverage waiting time %.2f\n",awt);
    printf("Average turnaround time %.2f\n",atat);

}
```

OUTPUT:



RESULT:

Thus the FIFO process scheduling program was executed and verified successfully.