Experiment No: 01

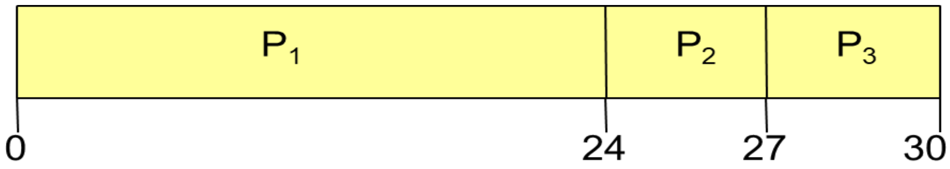
Experiment Name: Implementation of FCFS Algorithm Using C/C++.

FCFS Algorithm Theoretical Explanation:

Process	Burst Time	Arrival Time
P_1	24	0
P_2	3	0
P_3	3	0

With FCFS, the process that requests the CPU first is allocated the CPU first

Suppose that the processes arrive in the order: P_1, P_2, P_3.
The Gantt Chart for the schedule is:



Waiting time for $P_1 = 0$; $P_2 = 24$; $P_3 = 27$

Average waiting time: (0 + 24 + 27)/3 = 17

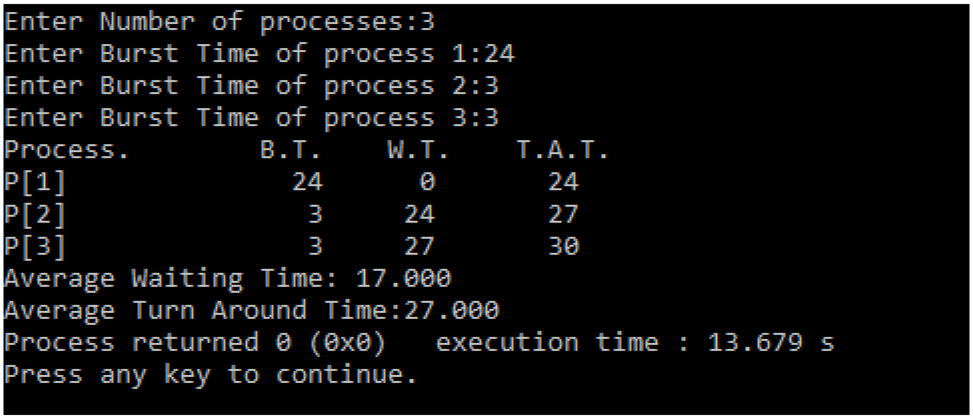
Average turn-around time: (24 + 27 + 30)/3 = 27

FCFS Algorithm Using C++ Code:

```
/* First Come First Serve CPU Scheduling Algorithm */
#include<bits/stdc++.h>
using namespace std;
int main ()
{
Int wt[10],bt[10],at[10],tat[10],n,i;
Float awt=0,atat=0;
printf("Enter Number of processes:");
scanf("%d",&n);
for(i=0; i<n; i++)
    {
printf("Enter Burst Time of process %d:",i+1);
scanf("%d",&bt[i]);
    }
wt[0]=0;
tat[0]=bt[0];
for(i=1; i<n; i++)
    {
wt[i]=bt[i-1]+wt[i-1];
awt+=wt[i];
tat[i]= wt[i]+bt[i];
atat+=tat[i];
    }
atat+=bt[0];
atat/=n;
awt/=n;

printf("Process.\tB.T.\tW.T.\tT.A.T.\n");
for(i=0; i<n; i++)
    {
printf("P[%d]\t        %3d\t%3d\t%4d\n",i+1,bt[i],wt[i],tat[i]);
    }
printf("Average Waiting Time: %0.3f\nAverage Turn Around Time:%0.3f",awt,atat);
return 0;
}
```

Output:



```
Enter Number of processes:3
Enter Burst Time of process 1:24
Enter Burst Time of process 2:3
Enter Burst Time of process 3:3
Process.        B.T.    W.T.    T.A.T.
P[1]              24      0       24
P[2]               3     24       27
P[3]               3     27       30
Average Waiting Time: 17.000
Average Turn Around Time:27.000
Process returned 0 (0x0)   execution time : 13.679 s
Press any key to continue.
```