Ex. No.: 6a) **Name:** Sarvesh
Date: 20/02/25 **Roll no.:** 230701294

FIRST COME FIRST SERVE

Aim:

To implement First-come First- serve (FCFS) scheduling technique

Algorithm:

1. Get the number of processes from the user.
2. Read the process name and burst time.
3. Calculate the total process time.
4. Calculate the total waiting time and total turnaround time for each process
5. Display the process name & burst time for each process. 6. Display the total waiting time, average waiting time, turnaround time

Program Code:

```
#include <stdio.h>
int main(){
int n;
printf("Enter the number of process:");
scanf("%d",&n);
int process[n],waiting[n],burst[n],turnaround[n];
for( int i=0;i<n;i++){
	printf("Enter burst time of process %d ",i);
	scanf("%d", &burst[i]);
}
waiting[0]=0;
int wsum=0;
for(int i=1;i<n;i++){
	waiting[i]= waiting[i-1]+burst[i-1];
wsum=wsum+waiting[i];
}
int tsum=0;
for(int i=0;i<n;i++){
	turnaround[i]=burst[i]+waiting[i];
	tsum=tsum+turnaround[i];
}
printf("Process\t\tBurst Time\t\tWaiting Time\t\tTurn around time\n");
for(int i=0;i<n;i++){
	printf("%d\t\t\t%d\t\t\t%d\t\t\t\t\t%d\n", i+1, burst[i],waiting[i], turnaround[i]);
}
printf("Avg waiting time %d\n", wsum/n);
printf("Avg turn around time %d\n", tsum/n);
```

Sample Output:

Enter the number of process:
3
Enter the burst time of the processes:
24 3 3

Process	Burst Time	Waiting Time	Turn Around Time
0	24	0	24
1	3	24	27
2	3	27	30

Average Waiting time is: 17.0
Average Turnaround Time is: 27.0

```
[student@localhost ~]$ cc fcfs1.c
[student@localhost ~]$ ./a.out
Enter the number of process:3
Enter burst time of process 0 24
Enter burst time of process 1 3
Enter burst time of process 2 3
Process         Burst Time              Waiting Time            Turn around time
0               24              0                       24
1               3               24                      27
2               3               27                      30
Avg waiting time 17
Avg turn around time 27
```

Result:

The FCFS program was executed and the output has been verified.