**EXPERIMENT-3**

3. Design a CPU scheduling program with C using First Come First Served technique with the

following considerations.

a. All processes are activated at time 0.

b. Assume that no process waits on I/O devices.

#include <stdio.h>

int main() {

int n, i;

int burstTime[100], waitingTime[100], turnaroundTime[100];

int totalWaitingTime = 0, totalTurnaroundTime = 0;

printf("Enter the number of processes: ");

scanf("%d", &n);

printf("Enter burst time for each process:\n");

for (i = 0; i < n; i++) {

printf("Process %d: ", i + 1);

scanf("%d", &burstTime[i]);

}

waitingTime[0] = 0;

for (i = 1; i < n; i++) {

waitingTime[i] = waitingTime[i - 1] + burstTime[i - 1];

}

for (i = 0; i < n; i++) {

turnaroundTime[i] = waitingTime[i] + burstTime[i];

totalWaitingTime += waitingTime[i];

totalTurnaroundTime += turnaroundTime[i];

}

printf("\nProcess\tBurst Time\tWaiting Time\tTurnaround Time\n");

for (i = 0; i < n; i++) {

printf("P%d\t%d\t\t%d\t\t%d\n", i + 1, burstTime[i], waitingTime[i], turnaroundTime[i]);

}

float avgWaitingTime = (float)totalWaitingTime / n;

float avgTurnaroundTime = (float)totalTurnaroundTime / n;

printf("\nAverage Waiting Time: %.2f\n", avgWaitingTime);

printf("Average Turnaround Time: %.2f\n", avgTurnaroundTime);

return 0;

}

**SAMPLE INPUT**

Enter the number of processes: 3

Enter burst time for each process:

Process 1: 5

Process 2: 3

Process 3: 8

**SAMPLE OUTPUT**

Process Burst Time Waiting Time Turnaround Time

P1 5 0 5

P2 3 5 8

P3 8 8 16

Average Waiting Time: 4.33

Average Turnaround Time: 9.67