NAME : Suryansh Pratap Singh

SECTION : K17FB

REGNO : 11702780

ROLL NO : 38

#include <stdio.h>

int main()

{

int arrival\_time[10], burst\_time[10], temp[10];

int i, smallest, count = 0, time, limit;

double wait\_time = 0, turnaround\_time = 0, end;

float average\_waiting\_time, average\_turnaround\_time;

printf("\nEnter the Total Number of Processes:\t");

scanf("%d", &limit);

printf("\nEnter Details of %d Processes\n", limit);

for(i = 0; i < limit; i++)

{

printf("\nEnter Arrival Time:\t");

scanf("%d", &arrival\_time[i]);

printf("Enter Burst Time:\t");

scanf("%d", &burst\_time[i]);

temp[i] = burst\_time[i];

}

burst\_time[9] = 9999;

for(time = 0; count != limit; time++)

{

smallest = 9;

for(i = 0; i < limit; i++)

{

if(arrival\_time[i] <= time && burst\_time[i] < burst\_time[smallest] && burst\_time[i] > 0)

{

smallest = i;

}

}

burst\_time[smallest]--;

if(burst\_time[smallest] == 0)

{

count++;

end = time + 1;

wait\_time = wait\_time + end - arrival\_time[smallest] - temp[smallest];

turnaround\_time = turnaround\_time + end - arrival\_time[smallest];

}

}

average\_waiting\_time = wait\_time / limit;

average\_turnaround\_time = turnaround\_time / limit;

printf("\n\nAverage Waiting Time:\t%lf\n", average\_waiting\_time);

printf("Average Turnaround Time:\t%lf\n", average\_turnaround\_time);

return 0;

}

Complexity of Program: O(n\*n)

Description of complexity:- Because I have used 3 for loops, one is nested and one is parallel loop that’s why complexity of the program is O(n\*n)

TestCase 1:-

Enter the Total Number of Processes: 5

Enter Details of 5 Processes

Enter Arrival Time: 0

Enter Burst Time: 2

Enter Arrival Time: 1

Enter Burst Time: 1

Enter Arrival Time: 2

Enter Burst Time: 5

Enter Arrival Time: 4

Enter Burst Time: 3

Enter Arrival Time: 6

Enter Burst Time: 2

Average Waiting Time: 1.600000

Average Turnaround Time: 4.200000

TestCase2:-

Enter the Total Number of Processes: 3

Enter Details of 3 Processes

Enter Arrival Time: 0

Enter Burst Time: 1

Enter Arrival Time: 3

Enter Burst Time: 1

Enter Arrival Time: 7

Enter Burst Time: 2

Average Waiting Time: 0.000000

Average Turnaround Time: 1.333333

TestCase 3:-

Enter the Total Number of Processes: 4

Enter Details of 4 Processes

Enter Arrival Time: 1

Enter Burst Time: 1

Enter Arrival Time: 2

Enter Burst Time: 2

Enter Arrival Time: 3

Enter Burst Time: 3

Enter Arrival Time: 4

Enter Burst Time: 4

Average Waiting Time: 1.000000

Average Turnaround Time: 3.500000

TestCase 4:-

Enter the Total Number of Processes: 3

Enter Details of 3 Processes

Enter Arrival Time: 3

Enter Burst Time: 1

Enter Arrival Time: 3

Enter Burst Time: 2

Enter Arrival Time: 3

Enter Burst Time: 3

Average Waiting Time: 1.333333

Average Turnaround Time: 3.333333

TestCase 5:-

Enter the Total Number of Processes: 3

Enter Details of 3 Processes

Enter Arrival Time: 1

Enter Burst Time: 1

Enter Arrival Time: 1

Enter Burst Time: 3

Enter Arrival Time: 2

Enter Burst Time: 5

Average Waiting Time: 1.333333

Average Turnaround Time: 4.333333