SCHOOL OF COMPUTING

GRAPHIC ERA HILL UNIVERSITY, BHIMTAL CAMPUS

2023-25

A

Term-Work

On

OPERATING SYSTEM LAB(PMC-102)

Submitted in partial fulfillment of the requirement for the Ist semester

M.C.A

By

Lalit Singh

237112182

…

Faculty-in-Charge

Mr. Praveen Joshi

Associate Professor

SCHOOL OF COMPUTING

GRAPHIC ERA HILL UNIVERSITY, BHIMTAL CAMPUS

2023-25

STUDENT'S DECLARATION

I **Lalit Singh** hereby declare the work, which is being presented in the term-work, entitled "OPERATING SYSTEM Lab" in partial fulfillment of the requirement for the award of the degree **M.C.A** in the session **2023-2025**, is an authentic record of my own work carried out under the supervision of **Mr. Praveen Joshi.**

The matter embodied in this term-work has not been submitted by me for the award of any other degree.

Date: …… …….

(Full signature of student)

Graphic Era
HILL UNIVERSITY
Established by an Act of the State Legislature of Uttarakhand (Adhiniyam Sankhya 12 of 2011)

BHIMTAL CAMPUS

INDEX

EXP. NO.	DATE	EXPERIMENT	PAGE NO.	REMARKS

PROGRAM NO.1

Name :-Lalit Singh
Course :-MCA
Subject :-Operating System Lab
Std id :-237112182
Date :-12 Aug 2023

Objective: To Implement the FCFS(First Come First Serve) CPU scheduling algorithm.

Code:

```
#include <stdio.h>
#include <stdlib.h>
struct process
{
 int at;
 int bt;
 int ct;
 int tat;
 int wt;
};
void find(struct process p[], int n)
{

 p[0].ct = p[0].at + p[0].bt;
 p[0].tat = p[0].ct - p[0].at;
 p[0].wt = p[0].tat - p[0].bt;
 for (int i = 1; i < n; i++)
 {
 p[i].ct = p[i - 1].ct + p[i].bt;
 p[i].tat = p[i].ct - p[i].at;
 p[i].wt = p[i].tat - p[i].bt;
 }

}
void display(struct process p[], int n)
{
 int i = 0;
 if (i == 0)
 {
 print("\nCompletion Time for Pocess%d:%d", i + 1, p[i].ct);
```

```
 printf("\nTurn Arou
nd Time for Pocess%d:%d", i + 1, p[i].tat);
 printf("\n Wait Time for Process%d:%d", i + 1, p[i].wt);
 }
 for (int i = 1; i < n; i++)
 {
 printf("\nCompletion Time for Pocess%d:%d", i + 1, p[i].ct);
 printf("\nTurn Around Time for Pocess%d:%d", i + 1, p[i].tat);
 printf("\n Wait Time for Process%d:%d", i + 1, p[i].wt);
 }
 Average_display(p, n);
}
void tablewise_display(struct process p[], int n)
{
 printf("\nProcess\tArrival         Time\tBurst         Time\tCompetion
Time\tTurnaround Time\tWaiting Time\n");
 for (int i = 0; i < n; i++)
 {
printf("%d\t%d\t\t%d\t\t%d\t\t%d\t\t%d\n",  i  +  1,  p[i].at,  p[i].bt,
p[i].ct, p[i].tat,
p[i].wt);
 }
Average_display(p, n);
}

void Average_display(struct process p[], int n){
float avg_wt = 0, avg_tat = 0;
for (int i = 0; i < n; i++) {
        avg_wt += p[i].wt;
        avg_tat += p[i].tat;
    }
    avg_wt /= n;
    avg_tat /= n;

printf("Average Waiting Time: %.2f\n", avg_wt);
printf("Average Turnaround Time: %.2f\n", avg_tat);
}
int main()
{
 int i, n,ch;
 printf("\nEnter the number of Processes: ");
 scanf("%d", &n);
 struct process p[n];
 for (i = 0; i < n; i++)
 {
 printf("\nEnter the arrival time for process %d:", i + 1);
```

```
 scanf("%d", &p[i].at);
 printf("\nEnter the burst time for process %d:", i + 1);
 scanf("%d", &p[i].bt);
 }

 while (1)
 {
 printf("\n1.Find\n2.Display\n3. Display in Tabular form\n4.Average tot
and wt\n5.Exit");
 printf("\n Enter your choice : ");
 scanf("%d", &ch);
 switch (ch)
 {
 case 1:
find(p, n);
 break;
 case 2:
 display(p, n);
 break;
 case 3:
 tablewise_display(p, n);
 break;
 case 4:
Average_display(p, n);
 break;
 case 5:
 exit(0);
 }
 }
}
```

Output:-

```
Enter the number of Processes: 3

Enter the arrival time for process 1:1

Enter the burst time for process 1:5

Enter the arrival time for process 2:2

Enter the burst time for process 2:8

Enter the arrival time for process 3:3
```

```
Enter the burst time for process 3:12

1.Find
2.Display
3.Display in Tabular form
4.Average tot and wt
5.Exit
 Enter your choice : 1

1.Find
2.Display
3.Display in Tabular form
4.Average tot and wt
5.Exit
 Enter your choice : 3

Process ArrivalTime  BurstTime CompetionTime TurnaroundTime Waiting Time
1       1              5              6              5              0
2       2              8              14             12             4
3       3              12             26             23             11
Average Waiting Time: 5.00
Average Turnaround Time: 13.33

1.Find
2.Display
3.Display in Tabular form
4.Average tot and wt
5.Exit
 Enter your choice : 4
Average Waiting Time: 5.00
Average Turnaround Time: 13.33
```