Operating System Lab

CEN-493

Program - 2

Code :-

```
#include <iostream>
using namespace std;

struct Process
{
    string pname;
    int arival_time;
    int burst_time;
    int waiting_time;
    int completion_time;
    int response_time;
    int turnaound_time;
};

void Print_Bars()
{
```

```
    for (int i = 0; i < 100; i++)
        cout << "_";
    cout << "\n";
}

void Insertion_Sort(Process Process_Array[], int
total_process)
{
    for (int i = 1; i < total_process; i++)
    {
        Process curent = Process_Array[i];
        int j = i - 1;
        while (Process_Array[j].arival_time >
curent.arival_time && j >= 0)
        {
            Process_Array[j + 1] = Process_Array[j];
            j--;
        }
        Process_Array[j + 1] = curent;
    }
}

void Average_Time(Process Process_Array[], int
total_process)
{
    double Av_CT = 0, Av_RT = 0, Av_WT = 0, Av_TAT = 0;
    for (int i = 0; i < total_process; i++)
    {
        Av_CT += Process_Array[i].completion_time;
        Av_RT += Process_Array[i].response_time;
        Av_TAT += Process_Array[i].turnaound_time;
        Av_WT += Process_Array[i].waiting_time;
    }
    Av_WT /= total_process;
    Av_TAT /= total_process;
    Av_RT /= total_process;
    Av_CT /= total_process;
    cout << "Average Time For The Different Time In
Process Scheduling\n\n";
```

```
    cout << "Average Completion Time -> " << Av_CT <<
"\n";
    cout << "Average Waiting Time -> " << Av_WT << "\n";
    cout << "Average Turn Around Time -> " << Av_TAT <<
"\n";
    cout << "Average Respond Time -> " << Av_RT << "\n";
}

void GanttChart(Process Process_Array[], int
total_process)
{

    cout << "Gantt Chart For Process Scheduling\n";
    cout << "\n";
    if (Process_Array[0].arival_time != 0)
    {
        cout << "|     |  ";
    }
    else
    {
        cout << "|  ";
    }

    for (int i = 0; i < total_process; i++)
    {
        if (i != 0 && Process_Array[i -
1].completion_time < Process_Array[i].arival_time)
        {
            cout << "    |   ";
        }
        cout << Process_Array[i].pname << "  |   ";
    }
    cout << "\n";

    if (Process_Array[0].arival_time != 0)
    {
        cout << " 0     ";
        cout << Process_Array[0].arival_time << "     ";
```

```
    }
    else
    {
        cout << Process_Array[0].arival_time << "      ";
    }

    for (int i = 0; i < total_process; i++)
    {
        if (i != 0 && Process_Array[i -
1].completion_time < Process_Array[i].arival_time)
        {
            cout << Process_Array[i].arival_time <<
"       ";
        }
        cout << Process_Array[i].completion_time <<
"       ";
    }
    cout << "\n";
}

void Chart(Process Process_Array[], int total_process)
{
    cout << "Various Time's Related To Process
Scheduling\n\n";
    cout <<
"|  Process   |  BT  |  AT  |   CT   |   WT  |  TAT  |  R
T  |\n";
    for (int i = 0; i < total_process; i++)
    {
        cout << "   " << Process_Array[i].pname << "\t\t"
<< Process_Array[i].burst_time
             << "\t" << Process_Array[i].arival_time <<
"\t" << Process_Array[i].completion_time
             << "\t" << Process_Array[i].waiting_time <<
"\t" << Process_Array[i].turnaound_time
             << "\t" << Process_Array[i].response_time <<
"\n";
    }
}
```

```
void FCFS(Process Process_Array[], int total_process)
{
    Insertion_Sort(Process_Array, total_process); //
Acording To A.T

    int timer = 0;
    for (int i = 0; i < total_process; i++)
    {
        if (timer < Process_Array[i].arival_time)
        {
            timer += (Process_Array[i].arival_time -
timer);
        }
        timer += Process_Array[i].burst_time;

        Process_Array[i].completion_time = timer;

        Process_Array[i].turnaound_time =
            Process_Array[i].completion_time -
            Process_Array[i].arival_time;

        Process_Array[i].waiting_time =
            Process_Array[i].turnaound_time -
            Process_Array[i].burst_time;

        Process_Array[i].response_time =
Process_Array[i].waiting_time;
    }
    Print_Bars();
    Chart(Process_Array, total_process);
    Print_Bars();
    Print_Bars();
    GanttChart(Process_Array, total_process);
    Print_Bars();
    Print_Bars();
    Average_Time(Process_Array, total_process);
    Print_Bars();
}
```

```
int main()
{
    system("cls");
    Print_Bars();
    cout << "20BCS070_Vicky_Gupta\n";
    cout << "First Come First Serve Process Scheduling
Alogorithm\n";
    Print_Bars();
    int total_process;
    cout << "Enter The No Of Processes : ";
    cin >> total_process;
    fflush(stdin);
    Process Process_Array[total_process];
    Print_Bars();
    cout << "Enter The Process Details...\n";
    cout << "| Process Name | Burst Time | Arival Time |
\n";

    for (int i = 0; i < total_process; i++)
    {
        cin >> Process_Array[i].pname;
        cin >> Process_Array[i].burst_time;
        cin >> Process_Array[i].arival_time;
    }

    FCFS(Process_Array, total_process);
    Print_Bars();
    cout << "Exited..\n";
    Print_Bars();
    return 0;
}
```

Output :-

```
-----------------------------------------------------------------------
20BCS070_Vicky_Gupta
First Come First Serve Process Scheduling Alogorithm
-----------------------------------------------------------------------
Enter The No Of Processes : 5
-----------------------------------------------------------------------
Enter The Process Details...
| Process Name | Burst Time | Arival Time |
P1      6       2
P2      2       5
P3      8       1
P4      3       0
P5      4       4
-----------------------------------------------------------------------
Various Time's Related To Process Scheduling

|  Process   |  BT  |  AT  |   CT   |   WT  |  TAT  |  RT  |
   P4          3       0       3       0       3       0
   P3          8       1       11      2       10      2
   P1          6       2       17      9       15      9
   P5          4       4       21      13      17      13
   P2          2       5       23      16      18      16
-----------------------------------------------------------------------
-----------------------------------------------------------------------
Gantt Chart For Process Scheduling

|  P4  |   P3  |   P1  |   P5  |   P2  |
0      3      11      17      21      23
-----------------------------------------------------------------------
-----------------------------------------------------------------------
Average Time For The Different Time In Process Scheduling

Average Completion Time -> 15
Average Waiting Time -> 8
Average Turn Around Time -> 12.6
Average Respond Time -> 8
-----------------------------------------------------------------------
-----------------------------------------------------------------------
Exited..
-----------------------------------------------------------------------
```