

OS PRACTICAL 5

Write a program that implements (with no preemption) scheduling algorithm.

CODE:

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

int main() {
    int total_process;
    int burst_time[30], waiting_time[30], turnaround_time[30];
    float average_waiting_time = 0.0, average_turnaround_time = 0.0;

    cout << "Enter the Number of Processes to Execute: ";
    cin >> total_process;

    cout << "Enter the Burst Time of Processes:\n";
    for (int count = 0; count < total_process; count++) {
        cout << "Process [" << count + 1 << "]: ";
        cin >> burst_time[count];
    }

    for (int i = 0; i < total_process - 1; i++) {
        for (int j = i + 1; j < total_process; j++) {
            if (burst_time[i] > burst_time[j]) {
                int temp = burst_time[i];
                burst_time[i] = burst_time[j];
                burst_time[j] = temp;
            }
        }
    }

    waiting_time[0] = 0;

    for (int count = 1; count < total_process; count++) {
```

```

        waiting_time[count] = waiting_time[count - 1] + burst_time[count - 1];
    }
    cout << "\nProcess\tBurst Time\tWaiting Time\tTurnaround Time\n";
    for (int count = 0; count < total_process; count++) {
        turnaround_time[count] = burst_time[count] + waiting_time[count];
        average_waiting_time += waiting_time[count];
        average_turnaround_time += turnaround_time[count];
        cout << "Process [" << count + 1 << "]\t"
            << burst_time[count] << "\t\t"
            << waiting_time[count] << "\t\t"
            << turnaround_time[count] << endl;
    }

    average_waiting_time /= total_process;
    average_turnaround_time /= total_process;

    cout << "\nAverage Waiting Time = " << fixed << setprecision(2) << average_waiting_time << endl;
    cout << "Average Turnaround Time = " << fixed << setprecision(2) << average_turnaround_time <<
endl;

    return 0;
}

```

OUTPUT :

```

C:\Users\shilp\OneDrive\Desktop>g++ 1.cpp && .\1.exe
Enter the Number of Processes to Execute: 3
Enter the Burst Time of Processes:
Process [1]: 3
Process [2]: 2
Process [3]: 5
Process Burst Time    Waiting Time    Turnaround Time
Process [1]           2                2
Process [2]           3                5
Process [3]           5               10
Average Waiting Time = 2.33
Average Turnaround Time = 5.67
-----
Process exited after 14.72 seconds with return value 0
Press any key to continue . . .

```