Experiment – 6

NAME : **Madhuram Brijeshkumar Modi**

ROLL NO. : **21BCP102**

DIV, GROUP : **2,G3**

Scheduling Algorithms

Scheduling Algorithms in OS:-

There are various algorithms which are used by the Operating System to schedule the processes on the processor in an efficient way.

The Purpose of a Scheduling algorithm:-

- Maximum CPU utilization
- Fare allocation of CPU
- Maximum throughput
- Minimum turnaround time
- Minimum waiting time
- Minimum response time

Different Type of algorithms: -

- First Come First Serve
- Round Robin
- Shortest Job First
- Shortest remaining time first
- Priority based scheduling
- Highest Response Ratio Next

1. First Come First Serve

It is the simplest algorithm to implement. The process with the minimal arrival time will get the CPU first. The lesser the arrival time, the sooner will the process gets the CPU. It is the non-preemptive type of scheduling.

```c
// write First Come First Serve algorithm asssuming each process arrives at time 0
#include <stdio.h>
#include <conio.h>
int main()
{
    int n, i, j;
    float avgwt = 0, avgtat = 0;
    printf("Enter the number of processes:");
    scanf("%d", &n);
    int bt[n], wt[n], tat[n];
    printf("Enter the burst time of the processes:");
    for (i = 0; i < n; i++)
    {
        scanf("%d", &bt[i]);
    }
    wt[0] = 0;
    tat[0] = bt[0];
    for (i = 1; i < n; i++)
    {
        wt[i] = wt[i - 1] + bt[i - 1];
        tat[i] = wt[i] + bt[i];
    }
    printf("Process\tBurst Time\tWaiting Time\tTurn Around Time");
    printf("\n");
    for (i = 0; i < n; i++)
    {
        printf("%d\t%d\t\t%d\t\t%d\n", i + 1, bt[i], wt[i], tat[i]);
        avgwt += wt[i];
        avgtat += tat[i];
    }
    avgwt /= n;
    avgtat /= n;
    printf("Average Waiting Time:%f", avgwt);
    printf("Average Turn Around Time:%f", avgtat);
}
```

OUTPUT

```
Enter the number of processes:5
Enter the burst time of the processes:4 3 5 2 1
Process Burst Time     Waiting Time     Turn Around Time
1       4              0                4
2       3              4                7
3       5              7                12
4       2              12               14
5       1              14               15
Average Waiting Time:7.400000Average Turn Around Time:10.400000

...Program finished with exit code 0
Press ENTER to exit console.
```

2. Shortest Job First

The job with the shortest burst time will get the CPU first. The lesser the burst time, the sooner will the process get the CPU. It is the non-preemptive type of scheduling.

```c
//write shortest job first algorithm assuming each process arrives at time 0
#include <stdio.h>
#include <conio.h>
int main()
{
    int n, i, j;
    float avgwt = 0, avgtat = 0;
    printf("Enter the number of processes:");
    scanf("%d", &n);
    int bt[n], wt[n], tat[n], p[n];
    printf("Enter the burst time of the processes:");
    for (i = 0; i < n; i++)
    {
        scanf("%d", &bt[i]);
        p[i] = i + 1;
    }
    for (i = 0; i < n; i++)
    {
        for (j = i + 1; j < n; j++)
        {
            if (bt[i] > bt[j])
            {
                int temp = bt[i];
                bt[i] = bt[j];
                bt[j] = temp;
                temp = p[i];
                p[i] = p[j];
                p[j] = temp;
            }
        }
    }
    wt[0] = 0;
    tat[0] = bt[0];
    for (i = 1; i < n; i++)
    {
        wt[i] = wt[i - 1] + bt[i - 1];
        tat[i] = wt[i] + bt[i];
    }
    printf("Process\tBurst Time\tWaiting Time\tTurn Around Time");
    printf("\n");
    for (i = 0; i < n; i++)
    {
        printf("%d\t%d\t\t%d\t\t%d\n", p[i], bt[i], wt[i], tat[i]);
        avgwt += wt[i];
        avgtat += tat[i];
    }
    avgwt /= n;
    avgtat /= n;
    printf("\n");
    printf("Average Waiting Time: %.2f\n", avgwt);
    printf("Average Turn Around Time: %.2f\n", avgtat);
}
```

OUTPUT-

```
Enter the number of processes:3
Enter the burst time of the processes:1 2 3
Process Burst Time      Waiting Time    Turn Around Time
1       1               0               1
2       2               1               3
3       3               3               6

Average Waiting Time: 1.33
Average Turn Around Time: 3.33


...Program finished with exit code 0
Press ENTER to exit console.
```