

OS LAB MANUAL

(CS23431)

Lab:3

Roll No:230701246

EX.NO:6b

SHORTEST JOB FIRST

Aim: To implement the Shortest Job First (SJF) scheduling technique

Program:

```
#include <stdio.h>
```

```
struct Process {
```

```
    int id;
```

```
    int burst_time;
```

```
    int waiting_time;
```

```
    int turnaround_time;
```

```
};
```

```
void sortProcesses(struct Process p[], int n) {
```

```
    struct Process temp;
```

```
    for (int i = 0; i < n - 1; i++) {
```

```
        for (int j = i + 1; j < n; j++) {
```

```
            if (p[i].burst_time > p[j].burst_time) {
```

```
                temp = p[i];
```

```
                p[i] = p[j];
```

```
                p[j] = temp;
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

```
int main() {
```

```

int n;

struct Process p[10];

int total_waiting_time = 0, total_turnaround_time = 0;

printf("Enter the number of processes: ");

scanf("%d", &n);

printf("Enter the burst time of the processes:\n");

for (int i = 0; i < n; i++) {

    printf("Process %d: ", i + 1);

    scanf("%d", &p[i].burst_time);

    p[i].id = i + 1;

    p[i].waiting_time = 0;

    p[i].turnaround_time = 0;

}

sortProcesses(p, n);

for (int i = 0; i < n; i++) {

    if (i == 0) {

        p[i].waiting_time = 0;

    } else {

        p[i].waiting_time = p[i - 1].waiting_time + p[i - 1].burst_time;

    }

    p[i].turnaround_time = p[i].waiting_time + p[i].burst_time;

    total_waiting_time += p[i].waiting_time;

    total_turnaround_time += p[i].turnaround_time;

}

printf("\nProcess\tBurst Time\tWaiting Time\tTurn Around Time\n");

for (int i = 1; i < n; i++) {

    printf("%d\t%d\t\t%d\t\t%d\n", p[i].id, p[i].burst_time, p[i].waiting_time,
p[i].turnaround_time);

}

printf("\nAverage waiting time is: %.2f", (float)total_waiting_time / n);

```

```
*printf("\nAverage Turn Around Time is: %.2f\n", (float)total_turnaround_time / n);

return 0;

}
```

Input:

```
praveen@LAPTOP-Q0D806DB:~$ vi sjf.c
praveen@LAPTOP-Q0D806DB:~$ gcc sjf.c
praveen@LAPTOP-Q0D806DB:~$ ./a.out
Enter the number of processes: 4
Enter the burst time of the processes:
Process 1: 2
Process 2: 3
Process 3: 1
Process 4: 4
```

OUTPUT:

```
praveen@LAPTOP-Q0D806DB:~$ vi sjf.c
praveen@LAPTOP-Q0D806DB:~$ gcc sjf.c
praveen@LAPTOP-Q0D806DB:~$ ./a.out
Enter the number of processes: 4
Enter the burst time of the processes:
Process 1: 2
Process 2: 3
Process 3: 1
Process 4: 4

Process Burst Time      Waiting Time      Turn Around Time
1         2              1                 3
2         3              3                 6
4         4              6                10

Average waiting time is: 2.50
Average Turn Around Time is: 5.00
```