

4ITRC2 Operating System Lab

Lab Assignment 5

To perform: Create and execute C programs for following CPU Scheduling Algorithms:

1. First Come First Serve (FCFS)

```
#include <stdio.h>
int main() {
    int n, bt[20], wt[20], tat[20];
    float avg_wt = 0, avg_tat = 0;

    printf("Enter number of processes: ");
    scanf("%d", &n);
    printf("Enter Burst Time for each process:\n");
    for (int i = 0; i < n; i++) {
        printf("P%d: ", i+1);
        scanf("%d", &bt[i]);
    }
    wt[0] = 0;
    for (int i = 1; i < n; i++)
        wt[i] = wt[i-1] + bt[i-1];

    for (int i = 0; i < n; i++) {
        tat[i] = wt[i] + bt[i];
        avg_wt += wt[i];
        avg_tat += tat[i];
    }
    printf("\nProcess\tBT\tWT\tTAT\n");
    for (int i = 0; i < n; i++)
        printf("P%d\t%d\t%d\t%d\n", i+1, bt[i], wt[i], tat[i]);
    printf("\nAverage Waiting Time: %.2f", avg_wt/n);
    printf("\nAverage Turnaround Time: %.2f\n", avg_tat/n);
    return 0;
}
```

OUTPUT:

P1 = 5, P2 = 9, P3 = 6

Avg WT = 6.33

Avg TAT = 13.00

2. Shortest Job First (SJF)

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

```
int main() {
```

```
    int n, bt[20], p[20], wt[20], tat[20];
```

```
    float avg_wt = 0, avg_tat = 0;
```

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

```
    scanf("%d", &n);
```

```
    for (int i = 0; i < n; i++) {  
        printf("Enter Burst Time for P%d: ", i+1);  
        scanf("%d", &bt[i]);  
        p[i] = i+1;  
    }
```

```
    // Sort processes by burst time
```

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

```
    for (int i = 1; i < n; i++)  
        wt[i] = wt[i-1] + bt[i-1];
```

```

for (int i = 0; i < n; i++) {
    tat[i] = wt[i] + bt[i];
    avg_wt += wt[i];
    avg_tat += tat[i];
}

printf("\nProcess\tBT\tWT\tTAT\n");
for (int i = 0; i < n; i++)
    printf("P%d\t%d\t%d\t%d\n", p[i], bt[i], wt[i], tat[i]);

printf("\nAverage Waiting Time: %.2f", avg_wt/n);
printf("\nAverage Turnaround Time: %.2f\n", avg_tat/n);

return 0;
}

```

OUTPUT:

Sorted: P1(5), P3(6), P2(9)

Avg WT = 5.33

Avg TAT = 12.00

3. Round Robin Scheduling

```

#include <stdio.h>

int main() {
    int n, tq, bt[10], rem_bt[10], wt[10] = {0}, tat[10], t = 0;
    float avg_wt = 0, avg_tat = 0;

    printf("Enter number of processes: ");
    scanf("%d", &n);

    printf("Enter Burst Time for each process:\n");
    for (int i = 0; i < n; i++) {
        printf("P%d: ", i+1);
        scanf("%d", &bt[i]);
        rem_bt[i] = bt[i];
    }

    printf("Enter Time Quantum: ");
}

```

```

scanf("%d", &tq);

int done;
do {
    done = 1;
    for (int i = 0; i < n; i++) {
        if (rem_bt[i] > 0) {
            done = 0;
            if (rem_bt[i] > tq) {
                t += tq;
                rem_bt[i] -= tq;
            } else {
                t += rem_bt[i];
                wt[i] = t - bt[i];
                rem_bt[i] = 0;
            }
        }
    }
} while (!done);

for (int i = 0; i < n; i++) {
    tat[i] = wt[i] + bt[i];
    avg_wt += wt[i];
    avg_tat += tat[i];
}

printf("\nProcess\tBT\tWT\tTAT\n");
for (int i = 0; i < n; i++)
    printf("P%d\t%d\t%d\t%d\n", i+1, bt[i], wt[i], tat[i]);

printf("\nAverage Waiting Time: %.2f", avg_wt/n);
printf("\nAverage Turnaround Time: %.2f\n", avg_tat/n);

return 0;
}

```

SAMPLE OUTPUT (TQ=4):

Avg WT = 10.67
Avg TAT = 17.33