**4ITRC2 Operating System Lab**

**Lab Assignment 5**

**Aim:** To create C programs for the different scheduling algorithms.

**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, i;

int bt[20], wt[20], tat[20];

float avg\_wt = 0, avg\_tat = 0;

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

scanf("%d", &n);

printf("Enter Burst Time for each process:\n");

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

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

scanf("%d", &bt[i]);

}

wt[0] = 0;

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

wt[i] = 0;

for(int j = 0; j < i; j++)

wt[i] += bt[j];

}

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

tat[i] = bt[i] + wt[i];

avg\_wt += wt[i];

avg\_tat += tat[i];

}

printf("\nProcess\tBT\tWT\tTAT\n");

for(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:-

Enter total number of processes: 3

Enter Burst Time for each process:

P1: 5

P2: 8

P3: 12

Process BT WT TAT

P1 5 0 5

P2 8 5 13

P3 12 13 25

Average Waiting Time = 6.00

Average Turnaround Time = 14.33

**2. Shortest Job First (SJF)**

#include <stdio.h>

int main() {

int n, bt[20], p[20], wt[20], tat[20], i, j, temp;

float avg\_wt = 0, avg\_tat = 0;

printf("Enter number of processes: ");

scanf("%d", &n);

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

printf("Enter Burst Time for P%d: ", i + 1);

scanf("%d", &bt[i]);

p[i] = i + 1;

}

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

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

if(bt[i] > bt[j]) {

temp = bt[i]; bt[i] = bt[j]; bt[j] = temp;

temp = p[i]; p[i] = p[j]; p[j] = temp;

}

}

}

wt[0] = 0;

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

wt[i] = 0;

for(j = 0; j < i; j++)

wt[i] += bt[j];

}

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

tat[i] = bt[i] + wt[i];

avg\_wt += wt[i];

avg\_tat += tat[i];

}

printf("\nProcess\tBT\tWT\tTAT\n");

for(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:-

Enter number of processes: 3

Enter Burst Time for P1: 6

Enter Burst Time for P2: 2

Enter Burst Time for P3: 8

Process BT WT TAT

P2 2 0 2

P1 6 2 8

P3 8 8 16

Average Waiting Time = 3.33

Average Turnaround Time = 8.67

**3. Round Robin Scheduling**

#include <stdio.h>

int main() {

int i, n, tq, bt[10], rt[10], wt[10] = {0}, tat[10] = {0};

int time = 0, remain;

float avg\_wt = 0, avg\_tat = 0;

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

scanf("%d", &n);

remain = n;

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

printf("Enter Burst Time for P%d: ", i + 1);

scanf("%d", &bt[i]);

rt[i] = bt[i];

}

printf("Enter Time Quantum: ");

scanf("%d", &tq);

while(remain != 0) {

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

if(rt[i] > 0) {

if(rt[i] > tq) {

time += tq;

rt[i] -= tq;

} else {

time += rt[i];

wt[i] = time - bt[i];

rt[i] = 0;

remain--;

}

}

}

}

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

tat[i] = bt[i] + wt[i];

avg\_wt += wt[i];

avg\_tat += tat[i];

}

printf("\nProcess\tBT\tWT\tTAT\n");

for(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:-

Enter total number of processes: 3

Enter Burst Time for P1: 5

Enter Burst Time for P2: 8

Enter Burst Time for P3: 6

Enter Time Quantum: 3

Process BT WT TAT

P1 5 9 14

P2 8 10 18

P3 6 12 18

Average Waiting Time = 10.33

Average Turnaround Time = 16.67