**SJF SCHEDULING**

#include <stdio.h>

int main() {

int n, i, j;

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

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

// Input number of processes

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

scanf("%d", &n);

// Input burst times

printf("Enter burst time for each process:\n");

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

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

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

p[i] = i + 1; // store process ID

}

// Sort processes by burst time (SJF)

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

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

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

// Swap burst time

int temp = bt[i];

bt[i] = bt[j];

bt[j] = temp;

// Swap process ID

temp = p[i];

p[i] = p[j];

p[j] = temp;

}

}

}

// First process waiting time is 0

wt[0] = 0;

// Calculate waiting time

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

wt[i] = 0;

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

wt[i] += bt[j];

}

}

// Calculate turnaround time and averages

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

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

avg\_wt += wt[i];

avg\_tat += tat[i];

}

avg\_wt /= n;

avg\_tat /= n;

// Print output

printf("\nProcess\tBurst Time\tWaiting Time\tTurnaround Time\n");

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

printf("P[%d]\t%d\t\t%d\t\t%d\n", p[i], bt[i], wt[i], tat[i]);

}

printf("\nAverage Waiting Time: %.2f", avg\_wt);

printf("\nAverage Turnaround Time: %.2f\n", avg\_tat);

return 0;

}