Assignment:3

**1. Write a C programme to simulate the following non-preemptive CPU scheduling algorithms to find the turnaround time and waiting time for the above problem.**

**FCFS, SJF, Priority Ans:**

#include <stdio.h>

void calculateTATandWT(int n, int bt[], int wt[], int tat[]) { wt[0] = 0;

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

wt[i] = bt[i - 1] + wt[i - 1]; for (int i = 0; i < n; i++)

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

}

void calculateAndDisplayAverageTATandWT(int n, int tat[], int wt[]) { int total\_tat = 0, total\_wt = 0;

for (int i = 0; i < n; i++) { total\_tat += tat[i]; total\_wt += wt[i];}

float avg\_tat = (float)total\_tat / n; float avg\_wt = (float)total\_wt / n;

printf("Average Turnaround Time = %.2f\n", avg\_tat); printf("Average Waiting Time = %.2f\n", avg\_wt);

}

void FCFS(int n, int bt[]) { int wt[n], tat[n];

calculateTATandWT(n, bt, wt, tat); printf("\nFCFS Scheduling\n");

printf("Process\tBurst Time\tWaiting Time\tTurnaround Time\n"); for (int i = 0; i < n; i++)

printf("%d\t%d\t\t%d\t\t%d\n", i + 1, bt[i], wt[i], tat[i]); calculateAndDisplayAverageTATandWT(n, tat, wt);

}

void SJF(int n, int bt[]) {

int wt[n], tat[n], temp; int sorted\_bt[n]; for (int i = 0; i < n; i++)

sorted\_bt[i] = bt[i];

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

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

if (sorted\_bt[i] > sorted\_bt[j]) { temp = sorted\_bt[i]; sorted\_bt[i] = sorted\_bt[j]; sorted\_bt[j] = temp;

}

}}

calculateTATandWT(n, sorted\_bt, wt, tat); printf("\nSJF Scheduling\n");

printf("Process\tBurst Time\tWaiting Time\tTurnaround Time\n"); for (int i = 0; i < n; i++)

printf("%d\t%d\t\t%d\t\t%d\n", i + 1, sorted\_bt[i], wt[i], tat[i]);

calculateAndDisplayAverageTATandWT(n, tat, wt);

}

void Priority(int n, int bt[], int priority[]) { int wt[n], tat[n], temp, temp\_p;

int sorted\_bt[n], sorted\_priority[n]; for (int i = 0; i < n; i++) {

sorted\_bt[i] = bt[i]; sorted\_priority[i] = priority[i];

}

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

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

if (sorted\_priority[i] > sorted\_priority[j]) { temp = sorted\_bt[i];

sorted\_bt[i] = sorted\_bt[j]; sorted\_bt[j] = temp; temp\_p = sorted\_priority[i];

sorted\_priority[i] = sorted\_priority[j]; sorted\_priority[j] = temp\_p;

}}}

calculateTATandWT(n, sorted\_bt, wt, tat); printf("\nPriority Scheduling\n");

printf("Process\tBurst Time\tPriority\tWaiting Time\tTurnaround Time\n"); for (int i = 0; i < n; i++)

printf("%d\t%d\t\t%d\t\t%d\t\t%d\n", i + 1, sorted\_bt[i], sorted\_priority[i], wt[i], tat[i]); calculateAndDisplayAverageTATandWT(n, tat, wt);

}

int main() {

int n;

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

int bt[n], priority[n];

printf("Enter the burst time of the processes:\n"); for (int i = 0; i < n; i++) {

printf("Process %d: ", i + 1); scanf("%d", &bt[i]);

}

FCFS(n, bt);

SJF(n, bt);

printf("Enter the priority of the processes:\n"); for (int i = 0; i < n; i++) {

printf("Process %d: ", i + 1); scanf("%d", &priority[i]);

}

Priority(n, bt, priority); return 0;

}

