Week 1

1. First Come First Serve

#include <stdio.h>

int main() {

int n, i;

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

scanf("%d", &n);

int bt[n], at[n], wt[n], tat[n];

int ct[n], total\_wt = 0, total\_tat = 0;

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

printf("Enter arrival time for Process %d: ", i + 1);

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

printf("Enter burst time for Process %d: ", i + 1);

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

}

ct[0] = at[0] + bt[0];

tat[0] = ct[0] - at[0];

wt[0] = tat[0] - bt[0];

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

if (ct[i-1] < at[i])

ct[i] = at[i] + bt[i];

else

ct[i] = ct[i-1] + bt[i];

tat[i] = ct[i] - at[i];

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

}

printf("Process\tArrival Time\tBurst Time\tWaiting Time\tTurnaround Time\n");

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

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

total\_wt += wt[i];

total\_tat += tat[i];

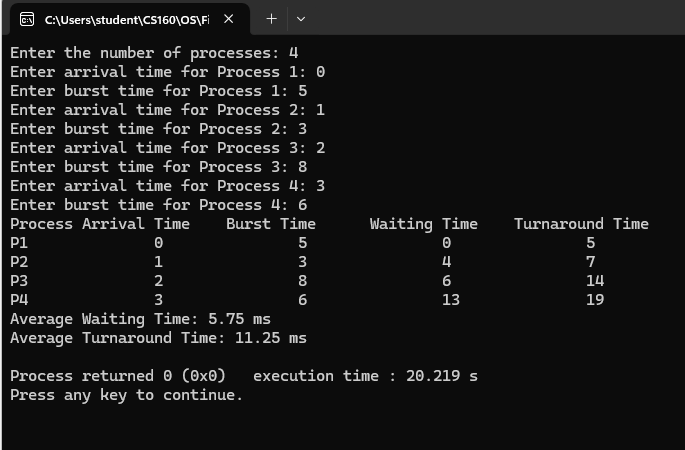
}

printf("Average Waiting Time: %.2f ms\n", (float)total\_wt / n);

printf("Average Turnaround Time: %.2f ms\n", (float)total\_tat / n);

return 0;

}



b) Shortest Job First

#include <stdio.h>

int main() {

int n, i, time = 0, count = 0, min\_bt, index;

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

scanf("%d", &n);

int bt[n], at[n], wt[n], tat[n], completed[n];

int total\_wt = 0, total\_tat = 0;

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

printf("Enter arrival time for Process %d: ", i + 1);

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

printf("Enter burst time for Process %d: ", i + 1);

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

completed[i] = 0; // Mark all processes as incomplete initially

}

while (count < n) {

min\_bt = 1e9;

index = -1;

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

if (at[i] <= time && !completed[i] && bt[i] < min\_bt) {

min\_bt = bt[i];

index = i;

}

}

if (index != -1) {

time += bt[index];

tat[index] = time - at[index];

wt[index] = tat[index] - bt[index];

completed[index] = 1;

count++;

} else {

time++;

}

}

printf("Process\tArrival Time\tBurst Time\tWaiting Time\tTurnaround Time\n");

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

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

total\_wt += wt[i];

total\_tat += tat[i];

}

printf("Average Waiting Time: %.2f\n", (float)total\_wt / n);

printf("Average Turnaround Time: %.2f\n", (float)total\_tat / n);

return 0;

}

