SJFS NON PREEMPETIVE

#include <stdio.h>

#include <stdlib.h>

#define NUM\_PROCESSES 4

typedef struct process

{

char name[3];

int burst\_time;

int completion\_time;

int waiting\_time;

int turnaround\_time;

} process;

int main()

{

process processes[NUM\_PROCESSES] =

{

{"P1", 6, 0, 0, 0},

{"P2", 8, 0, 0, 0},

{"P3", 7, 0, 0, 0},

{"P4", 3, 0, 0, 0}

};

for (int i = 0; i < NUM\_PROCESSES - 1; i++)

{

int min\_idx = i;

for (int j = i + 1; j < NUM\_PROCESSES; j++)

{

if (processes[j].burst\_time < processes[min\_idx].burst\_time)

{

min\_idx = j;

}

}

process temp = processes[i];

processes[i] = processes[min\_idx];

processes[min\_idx] = temp;

}

int completion\_time = 0;

int total\_waiting\_time = 0;

int total\_turnaround\_time = 0;

for (int i = 0; i < NUM\_PROCESSES; i++)

{

completion\_time += processes[i].burst\_time;

processes[i].completion\_time = completion\_time;

processes[i].turnaround\_time = completion\_time;

processes[i].waiting\_time = completion\_time - processes[i].burst\_time;

total\_waiting\_time += processes[i].waiting\_time;

total\_turnaround\_time += processes[i].turnaround\_time;

}

printf("Process Burst Time Completion Time Waiting Time Turnaround Time\n");

for (int i = 0; i < NUM\_PROCESSES; i++) {

printf("%s\t\t%d\t\t%d\t\t%d\t\t%d\n", processes[i].name, processes[i].burst\_time,

processes[i].completion\_time, processes[i].waiting\_time, processes[i].turnaround\_time);

}

printf("Average Waiting Time: %f\n", (float) total\_waiting\_time / NUM\_PROCESSES);

printf("Average Turnaround Time: %f\n", (float) total\_turnaround\_time / NUM\_PROCESSES);

return 0;

}

OUTPUT:

