#include<stdio.h>

int main()

{

int i, limit, total = 0, x, counter = 0, time\_quantum,j;

int wait\_time = 0, turnaround\_time = 0,pos,z,p[10],prio[10], arrival\_time[10], burst\_time[10], temp[10],b;

float average\_wait\_time, average\_turnaround\_time;

printf("\nEnter Total Number of Processes:");

scanf("%d", &limit);

x = limit;

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

{

p[i]=i+1;

prio[i]=0;

printf("\nEnter total Details of Process[%d]\n", i + 1);

printf("Arrival Time:\t");

scanf("%d", &arrival\_time[i]);

printf("Burst Time:\t");

scanf("%d", &burst\_time[i]);

temp[i] = burst\_time[i];

}

printf("\nEnter the Quantum/Time Slice :");

scanf("%d", &time\_quantum);

printf("Order of Execution is:");

printf("\nProcess ID\t\tBurst Time\tArrival time\tTurnaround Time\tWaiting Time\n");

for(total = 0, i = 0; x != 0;)

{

for(z=0;z<limit;z++)

{

int temp1;

pos=z;

for(j=z+1;j<limit;j++)

{

if(prio[j]<prio[pos])

pos=j;

}

temp1=prio[z];

prio[z]=prio[pos];

prio[pos]=temp1;

temp1=burst\_time[z];

burst\_time[z]=burst\_time[pos];

burst\_time[pos]=temp1;

temp1=arrival\_time[z];

arrival\_time[z]=arrival\_time[pos];

arrival\_time[pos]=temp1;

temp1=p[z];

p[z]=p[pos];

p[pos]=temp1;

temp1=temp[z];

temp[z]=temp[pos];

temp[pos]=temp1;

}

{

}

if(temp[i] <= time\_quantum && temp[i] > 0)

{

total = total + temp[i];

temp[i] = 0;

counter = 1;

}

else if(temp[i] > 0)

{

temp[i] = temp[i] - time\_quantum;

total = total + time\_quantum;

}

for(b=0;b<limit;b++)

{

if(b==i)

prio[b]+=1;

else

prio[b]+=2;

}

if(temp[i] == 0 && counter == 1)

{

x--;

printf("\nProcess[%d]\t\t%d\t\t %d\t\t %d\t\t%d", p[i], burst\_time[i],arrival\_time[i], total - arrival\_time[i], total - arrival\_time[i] - burst\_time[i]);

wait\_time = wait\_time + total - arrival\_time[i] - burst\_time[i];

turnaround\_time = turnaround\_time + total - arrival\_time[i];

average\_turnaround\_time=(float)turnaround\_time/(float)limit;

average\_wait\_time=(float)wait\_time/(float)limit;

counter = 0;

}

if(i == limit - 1)

{

i = 0;

}

else if(arrival\_time[i + 1] <= total)

{

i++;

}

else

{

i = 0;

}

}

printf("\nTotal Turn Around time: %d",turnaround\_time);

printf("\nTotal Waiting time:%d",wait\_time);

printf("\nAverage Turn Around Time: %f",average\_turnaround\_time);

printf("\nAverage Waiting Time:%f",average\_wait\_time);

return 0;

}