#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void doFCFS(char PName[50][10],int Detail[50][2],int n){

int ProcessWT[n],i,ProcessTAT[n];

ProcessWT[0] = 0;

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

if(i>0)

ProcessWT[i] = Detail[i-1][0] + ProcessWT[i-1];

ProcessTAT[i] = Detail[i][0] + ProcessWT[i];

}

printf("FCFS Scheduling\nProcess\_Name Burst Time Waiting Time Turn Around Time\n");

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

printf("%s\t\t%d\t\t%d\t\t%d\n",PName + i,Detail[i][0],ProcessWT[i],ProcessTAT[i]);

}

printf("\n");

}

void doPriority(char PName[50][10],int Detail[50][2],int n){

int ProcessWT[n],i,j,ProcessTAT[n];

ProcessWT[0] = 0;

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

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

if(Detail[j][1] > Detail[j+1][1]){

int tmp = Detail[j][1];

Detail[j][1] = Detail[j+1][1];

Detail[j+1][1] = tmp;

int tmp3 = Detail[j][0];

Detail[j][0] = Detail[j+1][0];

Detail[j+1][0] = tmp3;

char tmp2[10];

strcpy(tmp2,PName[j]);

strcpy(PName[j],PName[j+1]);

strcpy(PName[j+1],tmp2);

}

}

}

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

if(i>0)

ProcessWT[i] = Detail[i-1][0] + ProcessWT[i-1];

ProcessTAT[i] = Detail[i][0] + ProcessWT[i];

}

printf("Priority Scheduling\nProcess\_Name Burst Time Waiting Time Turn Around Time\n");

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

printf("%s\t\t%d\t\t%d\t\t%d\n",PName + i,Detail[i][0],ProcessWT[i],ProcessTAT[i]);

}

printf("\n");

}

void doRR(char PName[50][10],int Detail[50][2],int n){

int ProcessWT[n],remainingBT[n],i,ProcessTAT[n],quantum = 2;

ProcessWT[0] = 0;

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

remainingBT[i] = Detail[i][0];

}

int initTime = 0;

while(1){

int isDone = 0;

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

if(remainingBT[i] > 0){

isDone = 1;

if(remainingBT[i] > quantum){

initTime += quantum;

remainingBT[i] -= quantum;

}else{

initTime += remainingBT[i];

ProcessWT[i] = initTime - Detail[i][0];

remainingBT[i] = 0;

}

}

}

if(isDone==0)

break;

}

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

ProcessTAT[i] = Detail[i][0] + ProcessWT[i];

}

printf("Round Robin Scheduling\nProcess\_Name Burst Time Waiting Time Turn Around Time\n");

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

printf("%s\t\t%d\t\t%d\t\t%d\n",PName + i,Detail[i][0],ProcessWT[i],ProcessTAT[i]);

}

printf("\n");

}

void doSJF(char PName[50][10],int Detail[50][2],int n){

int ProcessWT[n],i,j,ProcessTAT[n];

ProcessWT[0] = 0;

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

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

if(Detail[j][0] > Detail[j+1][0]){

int tmp = Detail[j][0];

Detail[j][0] = Detail[j+1][0];

Detail[j+1][0] = tmp;

char tmp2[10];

strcpy(tmp2,PName[j]);

strcpy(PName[j],PName[j+1]);

strcpy(PName[j+1],tmp2);

}

}

}

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

if(i>0)

ProcessWT[i] = Detail[i-1][0] + ProcessWT[i-1];

ProcessTAT[i] = Detail[i][0] + ProcessWT[i];

}

printf("SJF Scheduling\nProcess\_Name Burst Time Waiting Time Turn Around Time\n");

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

printf("%s\t\t%d\t\t%d\t\t%d\n",PName + i,Detail[i][0],ProcessWT[i],ProcessTAT[i]);

}

printf("\n");

}

int main(){

int n,i,detail[50][2],opt;

char Pname[50][10];

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

scanf("%d",&n);

printf("Enter Process Name and Burst Time:");

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

scanf("%s %d",Pname[i],&detail[i][0]);

}

do{

printf("Select a Scheduling:\n1.FCFS\n2.SJF\n3.Round Robin(Pre-Emptive, Quantum = 2)\n4.Priority\n5.Exit\nEnter Option: ");

scanf("%d",&opt);

printf("\n");

switch(opt){

case 1:doFCFS(Pname,detail,n);

break;

case 2:doSJF(Pname,detail,n);

break;

case 3:doRR(Pname,detail,n);

break;

case 4:

printf("Enter Priority of each Process:");

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

printf("%s : ",Pname[i]);

scanf("%d",&detail[i][1]);

}

doPriority(Pname,detail,n);

break;

case 5:exit(0);

}

}while(1);

}