//Program for Election Algorithm - RING algorithm

#include<string.h>

#include<conio.h>

#include<stdio.h>

#include<stdlib.h>

struct rr

{

int index;

int id;

int f;

char state[10];

}proc[10];

int i,j,k,m,n;

void main()

{

int temp;

char str[10];

int init;

int ch;

int temp1;

int temp2;

int arr[10];

int max=-1;

clrscr();

printf("\n enter the number of process\t");

scanf("%d",&n);

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

{

//proc[i].index;

printf("\n enter priority of process %d \t",i);

scanf("%d",&proc[i].id);

strcpy(proc[i].state,"active");

proc[i].f=0;

}

// sorting

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

{

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

{

if(proc[j].id>proc[j+1].id)

{

temp=proc[j].id;

proc[j].id=proc[j+1].id;

proc[j+1].id=temp;

}

}

}

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

printf("[%d] %d\t",i,proc[i].id);

strcpy(proc[n-1].state,"inactive");

printf("\nprocess having priority %d selected

as coordinator",proc[n-1].id);

while(1)

{

printf("\n1)election 2)quit\n");

scanf("%d",&ch);

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

{

proc[i].f=0;

}

switch(ch)

{

case 1:

printf("\nenter the Process Number [i] who intialised election");

scanf("%d",&init);

temp2=init;

temp1=init+1;

i=0;

while(temp2!=temp1)

{

if(strcmp(proc[temp1].state,"active")==0 && proc[temp1].f==0 )

{

printf("process having priority %d sends a message to process

having priority %d \n",proc[init].id,proc[temp1].id);

proc[temp1].f=1;

init=temp1;

arr[i]=proc[temp1].id;

i++;

}

if(temp1==n)

temp1=0;

else

temp1++;

}

printf("process having priority %d sends a message to process

having priority %d \n",proc[init].id,proc[temp1].id);

arr[i]=proc[temp1].id;

i++;

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

{

if(max<arr[j])

max=arr[j];

}

printf("\nprocess with priority %d selected as coordinator",max);

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

{

if(proc[i].id==max)

{

strcpy(proc[i].state,"inactive");

printf("\n Cordinator process with priority %d deactivated\n",proc[i].id);

}

}

break;

case 2: exit(0);

}

}

}

/\*\*\*\*\*\*\*\*\*\*OUTPUT\*\*\*\*\*\*\*\*\*\*\*\*

enter the number of process 5

enter priority of process 0 78

enter priority of process 1 12

enter priority of process 2 56

enter priority of process 3 23

enter priority of process 4 45

[0] 12 [1] 23 [2] 45 [3] 56 [4] 78

process having priority 78 selected as coordinator

1)election 2)quit

1

enter the Process Number [i] who intialised election2

process having priority 45 sends a message to proces having priority 56

process having priority 56 sends a message to proces having priority 12

process having priority 12 sends a message to proces having priority 23

process having priority 23 sends a message to process having priority 45

process with priority 56 selected as coordinator

Cordinator process with priority 56 deactivated

1)election 2)quit

1

enter the Process Number [i] who intialised election1

process having priority 23 sends a message to proces having priority 45

process having priority 45 sends a message to proces having priority 12

process having priority 12 sends a message to process having priority 23

process with priority 45 selected as coordinator

Cordinator process with priority 45 deactivated

1)election 2)quit

2

\*/