Election Algorithm

#include<stdio.h>

#include<conio.h>

void bully()

{

int i,p[10],pno,ch,co,new\_co;

printf("\n\tHow many processes are in system : ");

scanf("%d",&pno);

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

{

p[i]=1;

}

do

{

printf("\n\t\*\*\*\*\* BULLY ALGO \*\*\*\*\*\n\t1.Crash Process\n\t2.Recover process\n\t3.Go in back menu\n\tEnter choice : ");

scanf("%d",&ch);

switch(ch)

{

case 1:printf("\n\tProcess %d is crash\*\*\*\*\*\n\tEnter co-ordinate to start election : ",pno);

p[pno-1]=0;

new\_co=0;

scanf("%d",&co);

l1 :

for(i=co;i<pno;i++)

printf("\n\t Election message to process %d",i+1);

for(i=co;i<pno;i++)

if(p[i]==1)

{

if(i>new\_co)

new\_co=i;

printf("\n\t Ok message from process %d ",i+1);

}

printf("\n\t\*\*\*\*New coordinator is %d \*\*\*\*\*",new\_co+1);

break;

case 2:

p[pno-1]=1;

co=pno-2;

goto l1;

break;

}

}while(ch!=3);

}

void ring()

{

int i,p[10],pno,ch,co,new\_co,flag=0;

printf("\n\tHow many processes are in system : ");

scanf("%d",&pno);

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

{

p[i]=1;

}

do

{

printf("\n\t\*\*\*\*\* RING ALGO \*\*\*\*\*\n\t1.Crash Process\n\t2.Recover process\n\t3.Go in back menu\n\tEnter choice : ");

scanf("%d",&ch);

switch(ch)

{

case 1:printf("\n\tProcess %d is crash\*\*\*\*\*\n\tEnter co-ordinate to start election : ",pno);

p[pno-1]=0;

scanf("%d",&co);

l2 :

new\_co=co;

for(i=co;i<pno;i++)

if(p[i])

{

printf("\n\t Message to %d ",i+1);

if(i>new\_co)

new\_co=i;

}

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

if(p[i])

printf("\n\t Message to %d ",i+1);

printf("\n\t\*\*\*\*Cycle completed New coordinator is %d \*\*\*\*\*",new\_co+1);

break;

case 2:

p[pno-1]=1;

co=pno-2;

goto l2;

break;

}

}while(ch!=3);

}

int main()

{

int ch;

clrscr();

do

{

printf("\n\t1.Bully \n\t2.Ring\n\t3.Exit\n\tEnter choice : ");

scanf("%d",&ch);

switch(ch)

{

case 1:

bully();

break;

case 2:

ring();

break;

}

}while(ch!=3);

}

Output: 





