**Selective Repeat ARQ** **protocol:-**

**CODE:-**

//Selective repeat sliding window protocol

#include<iostream>

using namespace std;

#include<conio.h>

#include<stdlib.h>

#include<time.h>

#include<math.h>

#define TOT\_FRAMES 500

#define FRAMES\_SEND 10

class sel\_repeat

{

private:

int fr\_send\_at\_instance;

int arr[TOT\_FRAMES];

int send[FRAMES\_SEND];

int rcvd[FRAMES\_SEND];

char rcvd\_ack[FRAMES\_SEND];

int sw;

int rw; //tells expected frame

public:

void input();

void sender(int);

void receiver(int);

};

void sel\_repeat::input()

{

int n; //no. of bits for the frame

int m; //no. of frames from n bits

int i;

cout<<"Enter the no. of bits for the sequence no. : ";

cin>>n;

m=pow(2,n);

int t=0;

fr\_send\_at\_instance=(m/2);

for(i=0;i<TOT\_FRAMES;i++)

{

arr[i]=t;

t=(t+1)%m;

}

for(i=0;i<fr\_send\_at\_instance;i++)

{

send[i]=arr[i];

rcvd[i]=arr[i];

rcvd\_ack[i]='n';

}

rw=sw=fr\_send\_at\_instance;

sender(m);

}

void sel\_repeat::sender(int m)

{

for(int i=0;i<fr\_send\_at\_instance;i++)

{

if(rcvd\_ack[i]=='n')

cout<<"SENDER : Frame "<<send[i]<<" is sent\n";

}

receiver(m);

}

void sel\_repeat::receiver(int m)

{

time\_t t;

int f;

int j;

int f1;

int a1;

char ch;

srand((unsigned)time(&t));

for(int i=0;i<fr\_send\_at\_instance;i++)

{

if(rcvd\_ack[i]=='n')

{

f=rand()%10;

//if f=5 frame is discarded for some reason

//else frame is correctly recieved

if(f!=5)

{

for(int j=0;j<fr\_send\_at\_instance;j++)

if(rcvd[j]==send[i])

{

cout<<"reciever:Frame"<<rcvd[j]<<"recieved correctly\n";

rcvd[j]=arr[rw];

rw=(rw+1)%m;

break;

}

int j;

if(j==fr\_send\_at\_instance)

cout<<"reciever:Duplicate frame"<<send[i]<<"discarded\n";

a1=rand()%5;

//if al==3 then ack is lost

//else recieved

if(a1==3)

{

cout<<"(acknowledgement "<<send[i]<<" lost)\n";

cout<<"(sender timeouts-->Resend the frame)\n";

rcvd\_ack[i]='n';

}

else

{

cout<<"(acknowledgement "<<send[i]<<" recieved)\n";

rcvd\_ack[i]='p';

}

}

else

{int ld=rand()%2;

//if =0 then frame damaged

//else frame lost

if(ld==0)

{

cout<<"RECEIVER : Frame "<<send[i]<<" is damaged\n";

cout<<"RECEIVER : Negative Acknowledgement "<<send[i]<<" sent\n";

}

else

{

cout<<"RECEIVER : Frame "<<send[i]<<" is lost\n";

cout<<"(SENDER TIMEOUTS-->RESEND THE FRAME)\n";

}

rcvd\_ack[i]='n';

}

}

}

for(int j=0;j<fr\_send\_at\_instance;j++)

{

if(rcvd\_ack[j]=='n')

break;

}

int i=0;

for(int k=j;k<fr\_send\_at\_instance;k++)

{

send[i]=send[k];

if(rcvd\_ack[k]=='n')

rcvd\_ack[i]='n';

else

rcvd\_ack[i]='p';

i++;

}

if(i!=fr\_send\_at\_instance)

{

for(int k=i;k<fr\_send\_at\_instance;k++)

{

send[k]=arr[sw];

sw=(sw+1)%m;

rcvd\_ack[k]='n';

}

}

cout<<"Want to continue";

cin>>ch;

cout<<"\n";

if(ch=='y')

sender(m);

else

exit(0);

}

int main()

{

sel\_repeat sr;;

sr.input();

//ducslectures.blogspot.in

}

**Output :-**

Enter the no of bits for the sequence number 6

SENDER : Frame 24 is sent

SENDER : Frame 32 is sent

SENDER : Frame 32 is sent

SENDER : Frame 27 is sent

SENDER : Frame 28 is sent

SENDER : Frame 29 is sent

SENDER : Frame 30 is sent

SENDER : Frame 31 is sent

RECIEVER : Frame 24 is lost

(SENDER TIMEOUTS --> RESEND THE FRAME)

RECIEVER : Frame 32 recieved correctly

(Acknowledgement 32 recieved)

RECIEVER : Frame 33 recieved correctly

(Acknowledgement 34 recieved)

RECIEVER : Frame 27 recieved correctly

(Acknowledgement 34 recieved)

RECIEVER : Frame 28 is damaged

RECIEVER : Negative acknowledgement 28 sent

RECIEVER : Frame 29 recieved correctly

(Acknowledgement 35 recieved)

RECIEVER : Frame 30 recieved correctly

(Acknowledgement 36 lost)

(SENDER TIMEOUTS --> RESEND THE FRAME)

RECIEVER : Frame 31 recieved correctly

(Acknowledgement 37 recieved)

Want to continue...n