```cpp
#include<iostream>
#include<stdlib.h>
#include<time.h>
#include<math.h>
#define TOT_FRAMES 500
#define FRAMES_SEND 10
using namespace std;
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, j, f1, 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;
```

```
67  //if f=5 frame is discarded for some reason
68  //else frame is correctly recieved
69                          if (f != 5)
70                          {
71                                  for (int j = 0; j < fr_send_at_instance; j++)
72                                          if (rcvd[j] == send[i])
73                                          {
74                                                  cout << "reciever:Frame" <<
    rcvd[j] << "recieved correctly\n";
75                                                  rcvd[j] = arr[rw];
76                                                  rw = (rw + 1) % m;
77                                                  break;
78                                          }
79                                  int j;
80                                  if (j == fr_send_at_instance)
81                                          cout << "reciever:Duplicate frame" <<
    send[i] << "discarded\n";
82                                  a1 = rand() % 5;
83  //if a1==3 then ack is lost
84  //else recieved
85                                  if (a1 == 3)
86                                  {
87                                          cout << "(acknowledgement " <<
    send[i] << " lost)\n";
88                                          cout << "(sender timeouts-->Resend
    the frame)\n";
89                                          rcvd_ack[i] = 'n';
90                                  }
91                                  else
92                                  {
93                                          cout << "(acknowledgement " <<
    send[i] << " recieved)\n";
94                                          rcvd_ack[i] = 'p';
95                                  }
96                          }
97                          else
98                          {       int ld = rand() % 2;
99  //if =0 then frame damaged
100 //else frame lost
101                                 if (ld == 0)
102                                 {
103                                         cout << " Receiver Side : Frame " <<
    send[i] << " is damaged\n";
104                                         cout << " Receiver Side : Negative
    Acknowledgement " << send[i] << " sent\n";
105                                 }
106                                 else
107                                 {
108                                         cout << "@Frame Receiver Side : " <<
    send[i] << " is lost\n";
109                                         cout << "(Sender Timeout-------->
    please resend frame)\n";
110                                 }
111                                 rcvd_ack[i] = 'n';
112                         }
113                 }
114         }
115         for (int j = 0; j < fr_send_at_instance; j++)
116         {
117                 if (rcvd_ack[j] == 'n')
118                         break;
119         }
120         int i = 0;
121         for (int k = j; k < fr_send_at_instance; k++)
122         {
123                 send[i] = send[k];
```

```
124                     if (rcvd_ack[k] == 'n')
125                             rcvd_ack[i] = 'n';
126                     else
127                             rcvd_ack[i] = 'p';
128                     i++;
129             }
130             if (i != fr_send_at_instance)
131             {
132                     for (int k = i; k < fr_send_at_instance; k++)
133                     {
134                             send[k] = arr[sw];
135                             sw = (sw + 1) % m;
136                             rcvd_ack[k] = 'n';
137                     }
138             }
139             cout << "do want to continue";
140             cin >> ch;
141             cout << "\n";
142             if (ch == 'y')
143                     sender(m);
144             else
145                     exit(0);
146     }
147     int main()
148     {
149             sel_repeat sr;
150             sr.input();
151     }
```