Ex:7

26.9.25

link layer

Aim : Write a program to implement flow control at data link layer using sliding window protocol. simulate the flow of frames from one mode to another.

Create a sender program with the following features :-

1) Input window size from the user
2) Input a text message from the user.
3) Consider 1 character per frame.
4) Create a frame with frame no and data.
5) send the frames.
6) Wait for the acknowledgement from the receiver.
7) Reader a file called Receiver_buffer.
8) check ACK field for the acknowledgement number.
9) If the acknowledgement number is expected and new set of frames, accordingly, else if NAC is found, resend the frames accordingly.

Create a receiver file with the following features :-

1) Reades a file called sender_buffer
2) check the frame no.
3) If the frame are as expected write the appropriate ACK no in the receiver buffer file else write NACK no. in the receiver buffer file else write NACK no in the receiver_buffer fide.

Sliding windows Protocol for sender:

```
#define SF "sender buffer.txt"
 void send_frames (int start, int end
                         char *msg)
 {
    file *f = fop(SF, "w");
    for (int i = ~~Start~~ Start ; i<end; i++)
      printf (f, "%d %c \n", i, msg[i]);
      fclose (f);
 }
 int main(){
     int win, base=0;
char msg[100];
printf (" windows size:");
Scanf ("%d", &win);
getchar();
int n = strlen(msg);
while (base <n){
int end = base + win;
send_frames (base, end, msg);
printf ("sent frames %d to %d\n",
                         base, end-1);
sleep(2);
Char type[10];
if (!get_ack(type&win)) continue;
else if (!strcmp (type, "NACK"))
                         base=win;
 }
 printf ("All frames sent \n");
}
```

Sliding window Protocol for receiver:

```
# define RF #receiver_buffer.txt"
  Void send_ack (const char*t, int n){
   FIFE *f = fopen (RF, "w");
   fprintf (f, "%s:%d", t, n);
   fclose(f);
}
```

```
int main(){
   int exp=0;
while (1){
   int f[100]; char d[100];
   int c = read_frames(fn, d);
   if (!c){
        sleep (1)
        continue;
   }
   for (int i=0; i<c; i++){
       if (fn[i]==exp){
          printf ("got frame %d %t\n",
                                  fn[i]);
                exp++; send_ack("ACK!")
   } else{
       printf ("wrong frame %d,
                          expected %d\n")
       send_ack ("NACK' exp);
            break;
   }
  }
  sleep (2);
 }
}
```

Sample Input :

```
Enter the no. of frames to send : 7
enter window size : 3
```

output :

```
sending frames : 012
Receiver : frame 0 received → sending ack0
Receiver : frame 1 received → sending ack1
Receiver : frame 2 received → sending ack2
```

Result :

Hence the sliding window protocol is completed and executed successfully