2) Write a Program to implement the data link layer farming methods such as i) Character stuffing ii) bit stuffing.

i) Character stuffing:

```c
#include<stdio.h>

#include<string.h>

main()

{

char a[30], fs[50] = " ", t[3], sd, ed, x[3], s[3], d[3], y[3];

int i, j, p = 0, q = 0;

clrscr();

printf("Enter characters to be stuffed:");

scanf("%s", a);

printf("\nEnter a character that represents starting delimiter:");

scanf(" %c", &sd);

printf("\nEnter a character that represents ending delimiter:");

scanf(" %c", &ed);

x[0] = s[0] = s[1] = sd;

x[1] = s[2] = '\0';

y[0] = d[0] = d[1] = ed;

d[2] = y[1] = '\0';

strcat(fs, x);

for(i = 0; i < strlen(a); i++)
```

```c
    {
     t[0] = a[i];
     t[1] = '\0';
     if(t[0] == sd)
     strcat(fs, s);
     else if(t[0] == ed)
     strcat(fs, d);
     else
     strcat(fs, t);
     }
     strcat(fs, y);
     printf("\n After stuffing:%s", fs);
     getch();
}
```

Output:-

Enter characters to be stuffed:goddmorning

Enter a character that represents starting delimiter:n

Enter a character that represents ending delimiter:g

 After stuffing: nggoddmornninnggg.

ii) Bit Stuffing:

```c
#include<stdio.h>

#include<string.h>

 int main()

{

int a[20],b[30],i,j,k,count,n;

printf("Enter frame size:");

scanf("%d",&n);

printf("Enter the frame in the form of0and1:")

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

scanf("%d",&a[i]);

 i=0;

count=1;

j=0;

while(i<n)

{

if(a[i]==1)

{

b[j]=a[i];

 for(k=i+1;a[k]==1 && k<n&&count<5;k++)

 {

 j++;

b[j]=a[k];
```

```
count++;

if(count==5)

{

j++;

b[j]=0;

 }

 }

 i=k

 }

 }

 else

 {

 b[j]=a[i];

 }

 i++;

 j++;

 }

printf("After Bit stuffing:");

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

printf("%d",b[i]);

   return 0;

   }
```

output

Enter frame size:8

Enter the frame in the form of 0 and 1:1 1 1 1 1 0 1 0

After Bit stuffing:1 1 1 1 1 0 0 1 0

3) Write a Program to implement data link layer farming method checksum.

```c
#include<stdio.h>

#include<math.h>

int sender(int arr[10],int n)

{

int checksum,sum=0,i;

printf("\n****SENDER SIDE****\n");

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

sum+=arr[i];

printf("SUM IS: %d",sum);

checksum=~sum;

printf("\nCHECKSUM IS:%d",checksum);

return checksum;

}

void receiver(int arr[10],int n,int sch)

{

int checksum,sum=0,i;

printf("\n\n****RECEIVER SIDE****\n");

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

sum+=arr[i];

printf("SUM IS:%d",sum);

sum=sum+sch;
```

```
checksum=~sum;   //1's

complement of sum

printf("\nCHECKSUMIS:%d",checksum);

 }

int main()

 {

     int n,sch,rch;

     printf("\nENTER SIZE OF THE STRING:");

     scanf("%d",&n);

     int arr[n];

     printf("ENTER THE ELEMENTS OF THE ARRAY TO CALCULATE
        CHECKSUM:\n");

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

     {

       scanf("%d",&arr[i]);

     }

    sch=sender(arr,n);

    receiver(arr,n,sch);

    }
```

OUTPUT:

ENTER SIZE OF THE STRING:2

ENTER THE ELEMENTS OF THE ARRAY TO CALCULATE CHECKSUM:

1001100

0110011

****SENDER SIDE****

SUM IS: 1111111

CHECKSUM IS:-1111112

****RECEIVER SIDE****

SUM IS:1111111

CHECKSUMIS:0

4)Write a program for Hamming Code generation for error detection and correction.

#include<stdio.h>

 void main()

 {

 int data[10];

int dataatrec[10],c,c1,c2,c3,i;

 printf("Enter 4 bits of data one by one\n");

 scanf("%d",&data[0]);

scanf("%d",&data[1]);

 scanf("%d",&data[2]);

 scanf("%d",&data[4]);

 data[6]=data[0]^data[2]^data[4];

 data[5]=data[0]^data[1]^data[4];

 data[3]=data[0]^data[1]^data[2];

printf("\nEncoded data is\n");

for(i=0;i<7;i++)

printf("%d",data[i]);

printf("\n\nEnter received data bits one by one\n");

 for(i=0;i<7;i++)

scanf("%d",&dataatrec[i]);

```
c1=dataatrec[6]^dataatrec[4]^dataatrec[2]^dataatrec[0];
c2=dataatrec[5]^dataatrec[4]^dataatrec[1]^dataatrec[0];
c3=dataatrec[3]^dataatrec[2]^dataatrec[1]^dataatrec[0];

 c=c3*4+c2*2+c1 ;

 if(c==0)

{

 printf("\nNo error while transmission of data\n");

}

 else

{

 printf("\nError on position %d",c);

 printf("\nData sent : ");

for(i=0;i<7;i++)

printf("%d",data[i]);

printf("\nData received : ");

for(i=0;i<7;i++)

printf("%d",dataatrec[i]);

printf("\nCorrect message is\n");

if(dataatrec[7-c]==0)

dataatrec[7-c]=1;

else
```

```
 dataatrec[7-c]=0;

for(i=0;i<7;i++)

{

 printf("%d",dataatrec[i]);

 }

 }

}
```

OUTPUT:

Enter 4 bits of data one by one

1

0

1

0

Encoded data is

1010010

Enter receivrd data bits one by one

1

0

1

0

0

1

0

No error while trnsmission of data

5) Write a Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.

```c
#include<stdio.h>

#include<math.h>

#include<string.h>

#define N strlen(g)

char t[28],cs[28],g[28];

int a,e,c,b;


void xor()

{

for(c=1;c<N;c++)

cs[c]=((cs[c]==g[c])?'0':'1');

}


void crc()

{

for(e=0;e<N;e++)

cs[e]=t[e];

do{

if(cs[0]=='1')

xor();

for(c=0;c<N-1;c++)

cs[c]=cs[c+1];

cs[c]=t[e++];
```

```
}while(e<=a+N-1);

}


int main()

{

int flag=0;

do{

printf("\n 1.CRC 12\n 2. CRC 16\n 3.CRC CCIP\n 4. EXIT\n\n Enter your option");

scanf("%d",&b);

switch(b)

 {

 case1:strcpy(g,"1100000001111");

 break;

 case2:strcpy(g,"11000000000000101");

 break;

 case3:strcpy(g,"10001000000100001");

 break;

 case4:return 0;

 }

printf("\nenter data:");

scanf("%s",t);

printf("\n----\n");

printf("generating polynomial:%s",g);

a=strlen(t);

for(e=a;e<a+N-1;e++)
```

```
t[e]='0';

printf("\n----\n");

printf("modified data is:%s",t);

printf("\n----\n");

crc();

printf("checksum is %s",cs);

for(e=a;e<a+N-1;e++)

t[e]=cs[e-a];

printf("\n----\n");

printf("\n final codeword is:%s",t);

printf("\n----\n");

printf("\n test error detection 0(yes) 1(no)?:");

scanf("%d",&e);

if(e==0)

 {

  do

  {

  printf("\n\t enter the position where error i to be inserted:");

   scanf("%d",&e);

  }while(e==0||e>a+N-1);

 t[e-1]=(t[e-1]=='0')?'1':'0';

 printf("\n---\n");

 printf("\n\t errorneous data:%s\n",t);

 }

crc();
```

```
for(e=0;(e<N-1)&&(cs[e]!='1');e++)

if(e<N-1)

printf("Error detected\n\n");

else

printf("no error detected\n\n");

printf("\n----\n");

}while(flag!=1);

}
```

OUTPUT:

1.CRC 12

2. CRC 16

3.CRC CCIP

4. EXIT

Enter your option1 enter data:111001

 ----

generating polynomial:

----

modified data is:111001

----

checksum is 0111111111111111111111111111111

----

final codeword
is:1110010111111111111111111111111111111111111111111111111111111

----

test error detection 0(yes) 1(no)?:0

enter the position where error i to be inserted:3

Errorneousdata:1100010111111111111111111111111111111111111111111111111111
error detected

----

6) Write a Program to implement Sliding window protocol for Goback N.

```
#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#include<unistd.h>

int n,r;

struct frame

{

char ack;

int data;

}frm[10];

int sender(void);

void recvack(void);

voidresend_gb(void);

int main()

{

int c;

sender();

recvack();

resend_gb();

printf("\nAll Frames sent successfully\n");
```

```
 }
int sender()
 {
int i;
printf("\n Enter no of Frames to be sent");
scanf("%d",&n); for(i=1;i<=n;i++)
{
printf("\n Enter data for Frames[%d]",i);
scanf("%d",&frm[i].data);
frm[i].ack='y';
 }
return 0;
 }
void recvack()
 {
int i; rand();
r=rand()%n;
frm[r].ack='n';
for(i=1;i<=n;i++)
{
if(frm[i].ack=='n')
printf("\n The frame number %d is not received",r);
```

```
} }

void resend_gb()

{

int i;

printf("\n Resending frame %d",r);

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

{

sleep(2);

frm[i].ack='y';

printf("\n The Received frame is %d",frm[i].data);

}

}
```

OUTPUT:

Enter no of Frames to be sent3

 Enter data for Frames[1]111

 Enter data for Frames[2]101

 Enter data for Frames[3]110

 The frame number 1 is not received

 Resending frame 1

 The Received frame is 111

 The Received frame is 101

 The Received frame is 110 All Frames sent successfully

7) Write a Program to implement Sliding window protocol for Selective repeat.

```c
#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#include<unistd.h>

int n,r;

struct frame

{

char ack;

int data;

}frm[10];

int sender(void);

void revack(void);

void resend_sr(void);

void selective(void);

int main()

{

sender();

revack();

resend_sr();

printf("\nAll Frames sent Successfully\n");

}

int sender()

{
```

```c
int i;

printf("\n Enter no.of Frames to be sent:");

scanf("%d",&n);

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

{

printf("\n Enter data for Frames [%d]",i);

scanf("%d",&frm[i].data);

frm[i].ack='y';

}

return 0;

}

void revack()

{

int i; rand();

r=rand()%n;

frm[r].ack='n';

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

{

if(frm[i].ack=='n')

printf("\n The Frame Number %d is not Received",i);

}

}

void resend_sr()

{

printf("\nResending Frame %d",r);
```

sleep(2); frm[r].ack='y';

printf("\n The Recieved Frame is %d",frm[r].data);

}

OUTPUT:

Enter no.of Frames to be sent:5

 Enter data for Frames [1]111

 Enter data for Frames [2]101

 Enter data for Frames [3]110

 Enter data for Frames [4]100

 Enter data for Frames [5]110

 The Frame Number 1 is not Received

Resending Frame 1

 The Recieved Frame is 111

All Frames sent Successfully

8) Write a Program to implement Stop and Wait Protocol.

```
#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#include<unistd.h>

int main()

{

int i,j,packet[30];

int fsize=(rand()%16)+1;

printf("\n\t Frame Size : %d\n",fsize);

printf("\n\t ----Data Log ----\n");

printf("\n FRAME\tDATA\tWaiting\tACKNOW\tRESEND");

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

{

packet[i]=rand()%1000;

printf("\n %d \t\t %d",i+1,packet[i]);

while(j=0||rand()%2==0)

{

sleep(1);

printf("\t1");

sleep(1);

for(j=2;rand()%2==0 && j<4;j++)
```

```
{

printf("%d",j);

sleep(1);

}

if(j==4)

printf("\t NO \t RESENDING...\n %d \t %d",i+1,packet[i]);

else break;

}

if(j==0)

{

sleep(1);

printf("\t0");

}

printf("\t YES \t NO");

}

printf("\n\n---- ALL DATA PACKETS SEND ----\n");

return 0;

}
```

OUTPUT:

    Frame Size : 8

     ----Data Log ----

 FRAME DATA   Waiting ACKNOW RESEND

```
1        886    0      YES    NO

2        915    0      YES    NO

3        335    12     YES    NO

4        421    1      YES    NO

5        690    0      YES    NO

6        763    123    NO     RESENDING...

6        763    1      YES    NO

7        368    0      YES    NO

8        429    123    NO     RESENDING...

8        429    0      YES    NO
```

---- ALL DATA PACKETS SEND ----

9) Write a program for congestion control using leaky bucket algorithm

```
#include<stdio.h>

#include<stdlib.h>

struct packet

{

int time;

int size;

}p[50];

int main()

{

int i,n,m,k=0; int bsize,bfilled,outrate;

printf("Enter the number of packets:");

scanf("%d",&n);

printf("Enter packets in the order of their arrival time");

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

{

printf("Enter the time and size:");

 scanf("%d%d",&p[i].time, &p[i].size);

}

printf("Enter the bucket size:");

scanf("%d",&bsize);

printf("Enter the output rate:");

scanf("%d",&outrate);

m=p[n-1].time;

i=1;
```

```
k=0;

bfilled=0;

while(i<=m||bfilled!=0)

{

printf("\n\nAt time %d",i);

if(p[k].time==i)

{

if(bsize>=bfilled+p[k].size)

{

bfilled=bfilled+p[k].size;

printf("\n%d bytes packet is inserted",p[k].size);

k=k+1;

}

else

{

printf("\n%d bytes packet is discarded",p[k].size);

k=k+1;

}

}

if(bfilled==0)

{

printf("\n No packets to transmit");

}

else if(bfilled>=outrate)

{
```

```
bfilled=bfilled-outrate;

printf("\n%d bytes transfered",outrate);

}
else
{
printf("\n%d bytes transfered",bfilled);

bfilled=0;

}
printf("\nPackets in the bucket %d bytes",bfilled);

i++;

}
return 0;

}
```

OUTPUT:

Enter the number of packets:1

Enter packets in the order of their arrival timeEnter the time and size:4 9

Enter the bucket size:2

Enter the output rate:4

At time 1

 No packets to transmit

Packets in the bucket 0 bytes

At time 2

 No packets to transmit

Packets in the bucket 0 bytes

At time 3

 No packets to transmit

Packets in the bucket 0 bytes

At time 4

9 bytes packet is discarded

 No packets to transmit

Packets in the bucket 0 bytes

10) Write a Program to implement Dijkstra's algorithm to compute the Shortest path through a graph.

```c
#include<stdio.h>

#include<conio.h>

#define INFINITY 9999

#define MAX 10

void dijkstra(int G[MAX][MAX],int n,int startnode);

int main()

{

int G[MAX][MAX],i,j,n,u;

printf("Enter no. of vertices:");

scanf("%d",&n);

printf("\nEnter the adjacency matrix:\n");

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

scanf("%d",&G[i][j]);

printf("\nEnter the starting node:");

scanf("%d",&u);

dijkstra(G,n,u);

return 0;

 }

void dijkstra(int G[MAX][MAX],int n,int startnode)

{
```

```
int cost[MAX][MAX],distance[MAX],pred[MAX];

 int visited[MAX],count,mindistance,nextnode,i,j;

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

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

 if(G[i][j]==0)

 cost[i][j]=INFINITY;

 else

 cost[i][j]=G[i][j];

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

 {

 distance[i]=cost[startnode][i];

 pred[i]=startnode;

 visited[i]=0;

 }

 distance[startnode]=0;

 visited[startnode]=1;

 count=1;

 while(count<n-1)

 {

 mindistance=INFINITY;

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

 if(distance[i]<mindistance&&!visited[i])
```

```
  {

    mindistance=distance[i];

     nextnode=i;

  }

  visited[nextnode]=1;

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

 if(!visited[i])

 if(mindistance+cost[nextnode][i]<distance[i])

 {

  distance[i]=mindistance+cost[nextnode][i];

   pred[i]=nextnode;

 }

 count++;

}

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

 if(i!=startnode)

 {

 printf("\nDistance of node%d=%d",i,distance[i]);

  printf("\nPath=%d",i);

  j=i;

  do {

  j=pred[j];
```

```
 printf("<%d",j);

 } while(j!=startnode);

}

}
```

Output:

Enter no. of vertices:6

Enter the adjacency matrix:

0 0 0 0 4 4

4 0 0 0 0 2

3 1 6 4 2 0

0 0 3 2 0 0

1 0 0 3 0 0

6 0 0 2 3 0

Enter the starting node:0

Distance of node1=10

Path=1<2<3<5<0

Distance of node2=9

Path=2<3<5<0

Distance of node3=6

Path=3<5<0

Distance of node4=4

Path=4<0

Distance of node5=4

Path=5<0

11) Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).

```c
#include<stdio.h>

 struct node

 {

 unsigned dist[20];

 unsigned from[20];

 }rt[10];

int main()

{

  int costmat[20][20];

  int nodes,i,j,k,count=0;

  printf("\nEnter the number of nodes : ");

  scanf("%d",&nodes);//Enter the nodes

  printf("\nEnter the cost matrix :\n");

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

  {

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

  {

  scanf("%d",&costmat[i][j]);

  costmat[i][i]=0;

  rt[i].dist[j]=costmat[i][j];//initialise the distance equal to cost matrix
```

```
  rt[i].from[j]=j;

  }

  }

  do {

  count = 0;

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

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

    for (k=0;k<nodes;k++)

    if(rt[i].dist[j]>costmat[i][k]+rt[k].dist[j]) ;

     {

     rt[i].dist[j]=rt[i].dist[k]+rt[k].dist[j];

      rt[i].from[j]=k;

      count++;

       }

  }while(count!=0);

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

  {

   printf("\n\n For router %d\n",i+1);

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

   {

    printf("\t\nnode %d via %d Distance %d ",j+1,rt[i].from[j]+1,rt[i].dist[j]);

   }

   }
```

```
printf("\n\n");

}
```

Output:

Enter the number of nodes : 3

Enter the cost matrix :

3 0 1

2 2 0

1 9 3

 For router 1

node 1 via 1 Distance 0

node 2 via 2 Distance 0

node 3 via 2 Distance 0

 For router 2

node 1 via 3 Distance 1

node 2 via 2 Distance 0

node 3 via 3 Distance 0

 For router 3

node 1 via 1 Distance 1

node 2 via 1 Distance 1

node 3 via 3 Distance 0

12) Write a Program to implement Broadcast tree by taking subnet of hosts.

```c
#include<stdio.h>

struct ed

{

int v1,v2,w;

}

edj[20],temp;

main()

{

int i,j,n=0,s,d,par[20],s1,d1;

clrscr();

printf("\n enter no of edges");

scanf("%d",&n);

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

{

printf("\n enter the node1,node2,seight");

scanf("%d%d%d",&edj[i].v1,&edj[i].v2,&edj[i].w);

par[i]=0;

}

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

{

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

if(edj[j].w>edj[i].w)
```

```
{

temp=edj[i];

edj[i]=edj[j];

edj[j]=temp;

}

}

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

printf("\n values %d\t%d\t%d\n",edj[i].v1,edj[i].v2,edj[i].w);

printf("\n BROADCAST TREE FOR THE GIVEN GRAPH\n");

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

{

s=edj[i].v1;

d=edj[i].v2;

s1=s;

d1=d;

while(par[s1]>0)

s1=par[s1];

while(par[d1]>0)

d1=par[d1];

if(s1!=d1)

{

par[d]=s;

printf("\n%d\t%d\t%d\n",s,d,edj[i].w);

}
```

}

}

OUTPUT:

enter no of edges 5

enter the node1,node2,seight

5

3

1

enter the node1,node2,seight

6

4

1

enter the node1,node2,seight

1

7

3

enter the node1,node2,seight 9

2

4

enter the node1,node2,seight

3

5

5

Values 5   3   1

Values 6   4   1

Values 1   7   3

values 9   2   4

values 3   5   5

BROADCAST TREE FOR

THE

GIVEN GRAPH

5   3   1