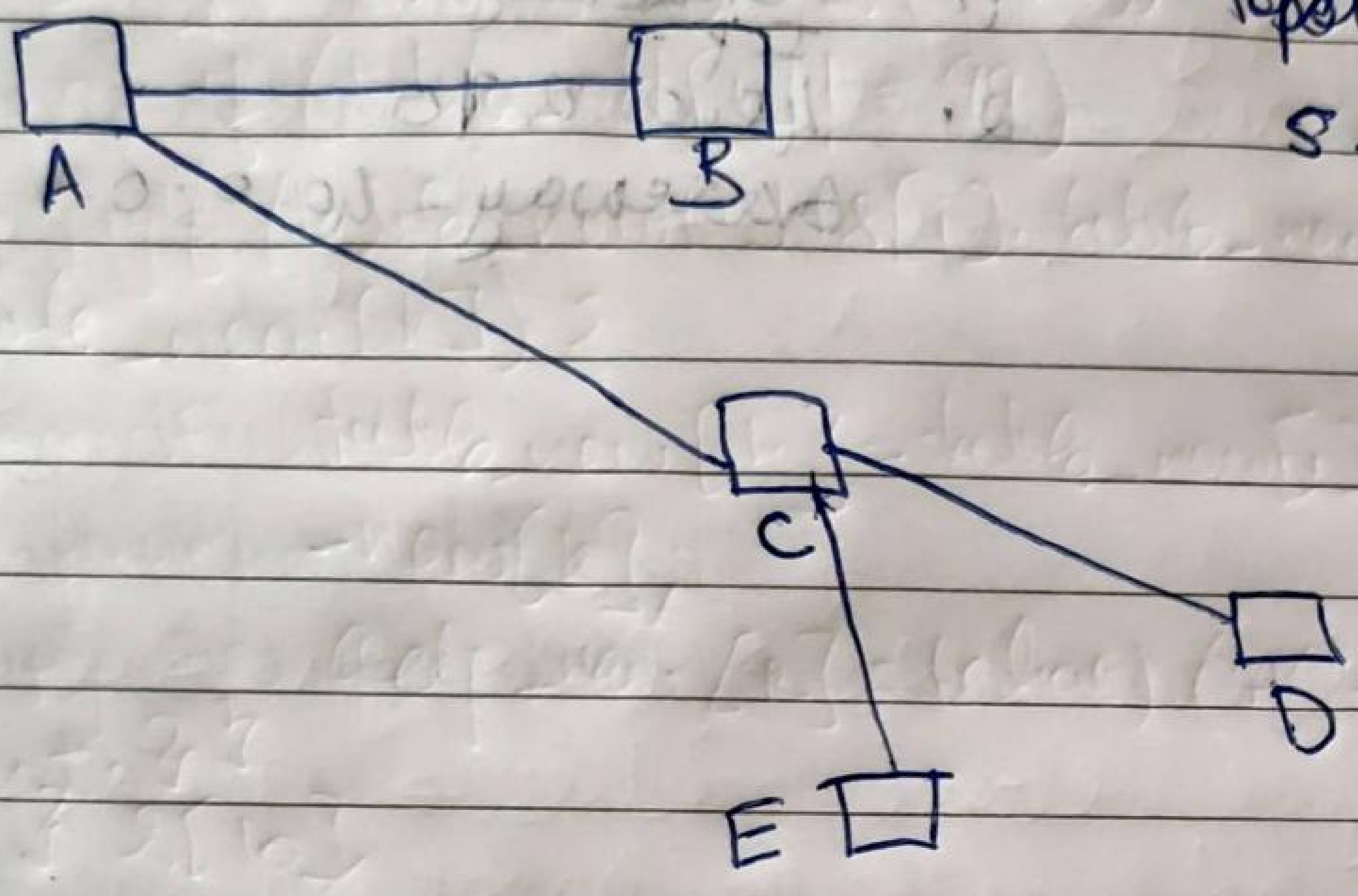
Distance Vector Algo



topology
5 routers

$$\begin{array}{c|ccccc} & A & B & C & D & E \\ \hline A & 0 & 1 & 1 & 0 & 0 \\ B & 1 & 0 & 0 & 0 & 0 \\ C & 1 & 0 & 0 & 1 & 1 \\ D & 0 & 0 & 1 & 0 & 0 \\ E & 0 & 0 & 1 & 0 & 0 \end{array}$$

at every router print routing table entries

header files

```
# define MAX 10
int n
class router {
char adj_new [MAX], adj_old[MAX];
int table_new [MAX], table_old[MAX];
public:
 router() {
  for (int i=0; i<MAX; i++) table_old[i]=
                          table_new[i]=99;
  }
```

classmate
Date
Page

```
void copy(){
for(i=0; i<n; i++){
    adj_old[i]=adj-new[i];
    table_old[i] = table-new[i];
  }
 }
int equal() {
 for(int i=0; i<n; i++)
  if ( table_old [i]!=table_new[i] || adj_new[i]!=adj-old[i])
    return 0;
   return 1;
 }

void input(int j) {
cout<<" Enter 1 if the corresponding router is adjacent to router" << (char) ('A'+j) <<" else enter 99 " <<
                                                        endl;

 for(i=0; i<n; i++)
    if (i!=j)
       cout <<(char) ('A' + 1) <<" ";
       cout<<"\n Enter matrix :";
  for(i=0; i<n; i++) {
       if (i==j)
               table_new[i]=0;
       else
            cin >> table_new [i];
            adj_new [i] =(char)('A' + i);
          }
          cout<<endl;
        }
        void display() {
        cout<<"\n Destination Router: ";
  for(i=0; i<n; i++) cout<< (char)('A'+ i)<<" ";
      cout<<"\n outgoing Line:
 for (i=0; i<n; i++) cout<< adj-new[i]<< " ";
     cout<<"\n Hop count: ";
```

```
for(i=0; i<n; i++)
  cout<<table_new[i]<<" ";
  }
void build (int j)
{
  for (i=0; i<n; i++)
    for(k=0; (i!=j) && (k<n); k++)
      if (table_old[i]!=99)
      if (table_new[i]+r[i].table_new[k]<
table_new[k]) {
           table_new[k]=table_new[i]+r[i].table
           -new[k];
           adj_new[k]=(char) ('A' + i);
     }}
  } r[10];

void build_table() {
i=0; j=0;
while (i!=n) {
for (i=j; i<n; i++) {
          r[i].copy();
          r[i].build(i);
     }
 for (i=0; i< n; i++)
   if (!r[i].equal()) {
             j=i;
             break; }
void main()
{
  cout<<"Enter no of routers (<"<<MAX<<"):
                                    "; cin >>
                                         n;
for(i=0; i<n; i++) r[i].input(i);
  build_table();
for(i=0; i<n; i++) { cout<<"Router table
entries for router"<<(char)('A'+i)<<":-";
r[i].display(); cout<<endl<<endl; } getch();
                                     }
```