Program 1

(1BM18CS086)

```
#include<conio.h>
#include<iostream.h>
#define MAX 10
int n;

class router {
  char adj_new[MAX] adj_old[MAX];
  int table_new[MAX], table_old[MAX];

  void copy()
  ct: ~~router()~~ {
      for(int 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 99: " <<endl<< " ";
```

```
for(int i=0;i<n;i++)
  if (i!=j)
    cout<<(char)('A'+i)<<" ";
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(int 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(int i=0; i<n; i++)
```

```
for(int 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[i0];

void build_table() {
  int 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;
    }
 }
}
```