This is a closed-book exam. Write the bodies of the functions `city::findmindist(int,char)` and `map::addhighway(char,char,int)`. Do not add any functions or make any attributes or functions public. The function `city::findmindist` should return the minimal distance between startcity and endcity if there is a path. The function `map::addhighway` should create the highway with the proper values and link it into the city class properly.

```cpp
#define MAXDIST 200
#define MAXROADS 40
#define MAXCITIES 20
using namespace std;
using namespace System;
class city;

// highways are one directional; that is, if there is a highway from city 1 to city 2, there is not automatically a highway from city 2 to city 1

class highway {
    city* startcity;
    city* endcity;
    int distance;

    public:
        highway(city* start, city* end, int dist){startcity = start;
            endcity = end; distance = dist;}
        city* getstart() {return startcity;}
        city* getend() {return endcity;}
        int getdist() {return distance;}
};

class city {
    char name;
    highway* roadsout[MAXROADS];
    int currentdist;
    int numroads;

    public:
        city(char newname) {name = newname; numroads = 0; currentdist = MAXDIST;}
        void addhighway(highway* newroad) {roadsout[numroads++] = newroad;}
        int findmindist(int dist, char destname);
        char getname() {return name;}
        void setdisthigh() {currentdist = MAXDIST;}
};

int findmindist(int dist, char destname){

}
```
class map {
    city* citylist[MAXCITIES];
    int numcity;
public:
    map(){numcity = 0;}
    void addcity(char newname){citylist[numcity++]=new city(newname);}
    city* findcity(char newname) {int i; for(i=0;i<numcity;i++)
        if(citylist[i]->getname()==newname){return citylist[i];}
        return 0;}
    void addhighway(char startcity, char endcity, int newdist);
    void setdisthigh() {int i; for(i=0;i<numcity;i++)
        {citylist[i]->setdisthigh();}}
    void findmindist(char startname, char endname) {int result;
        int dist = 0; city* start;
        start = findcity(startname); setdisthigh();
        result = start->findmindist(dist,endname);
        if(result == MAXDIST) {cout << "\n no path";}
        else
            {cout << startname << " " << endname << " " << result; }
    };
}

void addhighway(char startcity, char endcity, int newdist){