

```

#include <iostream>
#include <time.h>
using namespace std;
double cos(int x1, int y1, int x2, int y2)
{
    return sqrt((x1 - x2)*(x1 - x2) + (y1 - y2)*(y1 - y2));
}
int mn[5],z=0,pp[5][36];
void X(int i,int mn[5], int k)
{
    int j, a, m;
    if (i)
    {
        if (mn[i]>mn[i - 1])
        {
            m = i;
            for (i = k; i >= m; i--)
            {
                if (mn[i]>mn[m - 1])
                {
                    a = mn[m - 1];
                    mn[m - 1] = mn[i];
                    mn[i] = a;
                    for (j = k, i = m; j>i; j--, i++)
                    {
                        a = mn[i];
                        mn[i] = mn[j];
                        mn[j] = a;
                    }
                    for (i = 0; i <= k; i++){
                        cout << " " << mn[i];
                    }
                    if (mn[0] == 5){
                        for (i = 0; i < 5; i++){
                            pp[i][z] = mn[i];
                        }
                        z++;
                    }
                    //cout << "\n";
                    break;
                }
            }
        }
        i--;
        X(i, mn, k);
    }
}

void main()
{
    int p[10][10], i, y;
    float k[5][5],sum;
    for (i = 0; i < 5; i++)
    {
        p[i][0] = i + 1;
        for (y = 0; y < 5; y++)
            k[i][y] = 0;
    }
    p[3][4] = 1;
    p[4][7] = 1;
    p[2][3] = 1;
    p[6][9] = 1;
    p[7][4] = 1;
    k[1][0] = cos(3, 4, 4, 7);
}

```

```

k[2][0] = cos(3, 4, 2, 3);
k[3][0] = cos(3, 4, 6, 9);
k[4][0] = cos(3, 4, 7, 4);

k[2][1] = cos(4, 7, 2, 3);
k[3][1] = cos(4, 7, 6, 9);
k[4][1] = cos(4, 7, 7, 4);

k[3][2] = cos(2, 3, 6, 9);
k[4][2] = cos(2, 3, 7, 4);
k[4][3] = cos(6, 9, 7, 4);

for (i = 0; i < 5; i++)
for (y = i + 1; y < 5; y++)
{
    k[i][y] = k[y][i];
    cout << k[y][i] << endl;
    cout << k[i][y] << endl;
}

for (i = 0; i <= 4; i++)
{
    mn[i] = i + 1;
}
X(4, mn, 4);
cout << "All path";
for (i = 0; i < 24; i++){
    for (sum=0,y = 1; y < 5; y++)
    {
        cout << pp[y][i];
        sum += k[pp[y][i]-1][pp[y - 1][i]-1];
    }
    cout <<sum <<endl;

}

//
/*srand(time(NULL));
bool l[50][20];
int h, hp = 0, f, s, c = 0, cr;
do
{
    c++;
    for (i = 0; i < 24; i++)
    {
        for (sum = 0, y = 1; y < 5; y++)
            h += k[pp[y][i] - 1][pp[y - 1][i] - 1];

        for (y = 1; y < 5; y++)
            hp += k[pp[y][0] - 1][pp[y - 1][0] - 1];

        for (int u = 1; u < 100; u++)
            for (i = 1; i < 24; i++)
            {
                if (hp == 20 && h == 99){
                    cout << "ideal complete!!\n";
                    exit(0);
                }
            }
        for (hp=0,y = 1; y < 5; y++)
        {

```

```

        hp += k[pp[y][i] - 1][pp[y - 1][i] - 1];
    }
    if (h < hp)
    {
        for (y = 0; y < 5; y++)
        {
            l[0][y] = pp[i][y];
            pp[i][y] = pp[i - 1][y];
            pp[i - 1][y] = l[0][y];
        }
        hp = h;
    }
    for (i = 0; i < 12; i++)
    {
        f = rand() % 5;
        s = rand() % 5;
        for (y = 0; y < 5; y++)
            if (pp[24][y])
                l[i][y] = pp[f][y];
            else
                l[i][y] = pp[s][y];
    }
    for (i = 0; i < 12; i++){
        for (y = 0; y < 5; y++){
            pp[i + 12][y] = l[i][y];
        }
    }
    for (i = 0; i < 24; i++)
        for (y = 1; y < 5; y++)
        {
            h += k[pp[y][i] - 1][pp[y - 1][i] - 1];
        }

    for (hp = 0, y = 1; y < 5; y++)
        hp += k[pp[y][0] - 1][pp[y - 1][0] - 1];
    cout << hp << endl << h / 24 << "-----
-----" << endl;
} while (true);*/
}

```