最优规划

question 1

某工厂每日 8小时的产量不低于 1800 件。为了进行质量控制，计划聘请两个不同水平的检验员。一级检验员的速度为 25 件/小时，正确率 98%，计时工资 4元/小时，二级检验员的速度为 15件/小时，正确率 95%，计时工资 3元/小时，检验员每错一次，工厂要损失 2 元。现有可供厂方聘请的检验员人数为一级 7人和二级 8人。为使总检验费用最省，该工厂应聘请一级、二级检验员各多少名？

解：可设需要一级和二级检验员的人数分别为 $x1$ 和 $x2$ 名，根据题意，可得matlab方程：

```
clc
clear
f = [40;36];
A = [1 0
     0 1
     -5 -3];
b = [7;8;-45];
lb = zeros(2,1);
[x,fval,exitflag,output,lambda] = linprog(f,A,b,[],[],lb);
```

解得：

```
x =

    7.0000
    3.3333


fval =

  400.0000


exitflag =

     1


output =

         iterations: 6
          algorithm: 'interior-point'
       cgiterations: 0
            message: 'Optimization terminated.'
    constrviolation: 0
      firstorderopt: 4.9052e-07
```

可见招聘一级检验员7名，二级检验员3名可使总检验费用最少，约为400.00元

question 2

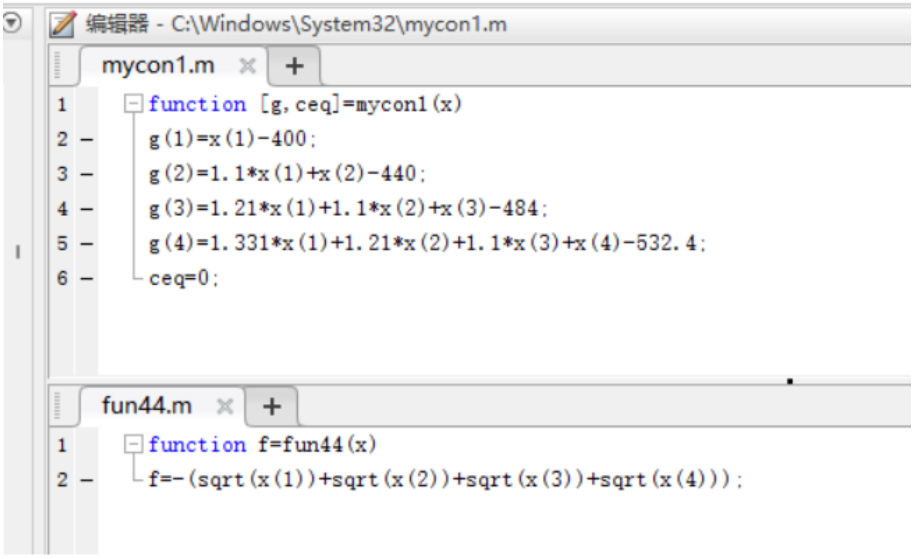
设有400万元资金，要求4年内使用完，若在一年内使用资金 x万元，则可得效益根号 x 万元（效益不能再使用），当年不用的资金可存入银行，年利率为10%。试制定出资金的使用计划，以使4年效益之和为最大。

解：

创建文件fun44.m和mycon1.m两个文件

```
function f=fun44(x)
f=-(sqrt(x(1))+sqrt(x(2))+sqrt(x(3))+sqrt(x(4)));
```

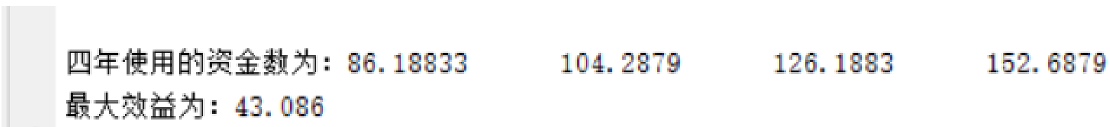
```
function [g,ceq]=mycon1(x)
g(1)=x(1)-400;
g(2)=1.1*x(1)+x(2)-440;
g(3)=1.21*x(1)+1.1*x(2)+x(3)-484;
g(4)=1.331*x(1)+1.21*x(2)+1.1*x(3)+x(4)-532.4;
ceq=0;
```



主程序：

```
x0=[1;1;1;1];
vlb=[0;0;0;0];
vub=[];
A=[];
b=[];
Aeq=[];
beq=[];
[x,fval]=fmincon('fun44',x0,A,b,Aeq,beq,vlb,vub,'mycon1');
disp(['四年使用的资金数为：',num2str(x')]);
disp(['最大效益为：',num2str(-fval)]);
```

可得结果：



question 3

一个商人欲到 n 个城市推销商品，每两个城市 i 和 j 之间的距离为 d_{ij} ，如何选择一条道路使得商人每个城市走一遍后回到起点且所走路径最短。

解：

用一个例子解决问题：

从城市1出发经过所有城市后回到城市1，要使总路程最短。

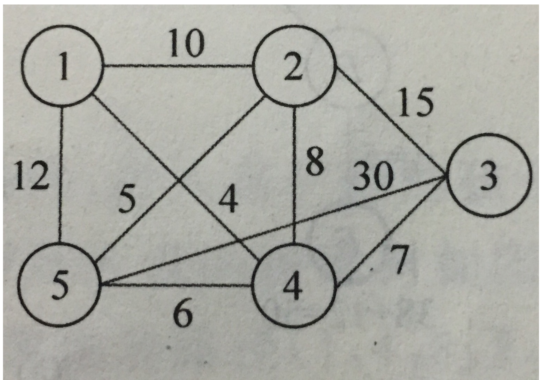


图 5-35 无向带权图

给定 n 个城市的无向带权图 $G(V,E)$，顶点代表城市，权值代表城市之间的距离。若城市之间没有路径，则距离为无穷。城市之间的距离存放在二维数组 $g[\quad][\quad]$中。从城市 1 出发，先到临近城市 2，将走过的路程存放在变量 cl 中。$bestl$代表当前找到的一种最短路径长度。如走法：$1 \to 2 \to 3 \to 4 \to 5 \to 1$。显然，向城市深处走时，$cl$ 只会增加。因此当 $cl > bestl$ 时，不必再往深处走。限界条件为 $cl < bestl$，cl 初值为0，$bestl$ 初值为 ∞。

程序:

```
#include<iostream>
using namespace std;
#define MAX 1000
int g[100][100], x[100], bestx[100];

int cl = 0, bestl = MAX, n;

//界定函数，按照屈婉玲老师的算法公开课视频讲解的给出的实现，这与网上绝大多数版本的界定函数不同，可以更精确的削减分支
double Bound(int t, int cl)
{
    double min1 = 0, min2 = 0, tempSum=0;
    for (int j = t; j <= n; j++)
    {
        if (g[x[t - 1]][x[j]] != -1 && g[x[t - 1]][x[j]] < min1)
        {
            min1 = g[x[t - 1]][x[j]];
        }
        for (int i = 1; i <= n; ++i)
        {
            if (g[x[j]][x[i]] != -1 && g[x[j]][x[i]] < min2)
            {
                min2 = g[x[j]][x[i]];
            }
        }
        tempSum += min2;
    }

    return cl + min1 + tempSum;

}


void Traveling(int t)
{
    int j;
    if (t>n) //到达叶子结点
    {

        if (g[x[n]][1] != -1 && (cl + g[x[n]][1]<bestl))//推销员到的最后一个城市与出发的城市之间有路径，且当前总距离比当前最优值小
        {
            for (j = 1; j <= n; j++)
                bestx[j] = x[j];
```

```
                bestl = cl + g[x[n]][1];
            }
        }
        else    //没有到达叶子结点
        {
            for (j = t; j <= n; j++)//搜索扩展结点的左右分支，即所有与当前所在城市临近的城市
            {
                if (g[x[t - 1]][x[j]] != -1 && Bound(t, cl)<bestl)
                //if (g[x[t - 1]][x[j]] != -1 && (cl + g[x[t - 1]][x[j]]<bestl))//若果第t-1个城市与第t个城市之间有路径且可以得到更短的路线
                {

                    swap(x[t], x[j]);     //保存要去的第t个城市到x[t]中
                    cl += g[x[t - 1]][x[t]]; //路线长度增加
                    Traveling(t + 1);      //搜索下一个城市
                    cl -= g[x[t - 1]][x[t]];
                    swap(x[t], x[j]);
                }
            }
        }
}
int main()
{
    int i, j;
    cout << "请输入一共有几个城市：" << endl;
    cin >> n;
    cout << "请输入城市之间的距离" << endl;

    for (i = 1; i <= n; i++)
        for (j = 1; j <= n; j++)
            cin >> g[i][j];

    for (i = 1; i <= n; i++)
    {
        x[i] = i;
        bestx[i] = 0;
    }

    Traveling(2);
    cout << "城市路线：" << endl;
    for (i = 1; i <= n; i++)
        cout << bestx[i] << ' ';
    cout << bestx[1];
    cout << endl;
    cout << "最短路线长度：" << endl;
    cout << bestl << endl;
    return 0;
}
```

测试输入:

```
4
0 30 6 4
30 0 5 10
6 5 0 20
4 10 20 0
```

结果:

question 4

用直径为1的圆木制作截面是矩形的梁，为使重量最轻而且强度最大，问截面的长和宽应取何尺寸?

解:

根据经验，梁的强度与高度平方成正比，和宽度成反比，可以写成 $q = y^2x$，又有 $x^2 + y^2 = 1$，得到

$$q = (1 - x^2)x = x - x^3$$

而导数 $q' = 1 - 3x^2$，当 $q' = 0$ 时，q 有极值，$q' = 0$ 时，$x = \frac{1}{\sqrt{3}} = 0.577$，此时 $y = \frac{\sqrt{6}}{3}$，有

$$x : y = 1 : \sqrt{2}$$

question 5

某厂生产一种产品有甲、乙两个牌号，讨论在产销平衡的情况下如何确定各自的产量，使总利润最大。所谓产销平衡指工厂的产量等于市场上的销量。

解:

符号说明:

$z(x_1, x_2)$ 表示总利润；p_1，q_1，x_1分别表示甲的价格、成本、销量；p_2，q_2，x_2分别表示乙的价格、成本、销量；
a_{ij}，b_i，$\lambda_i, c_i(i, j = 1, 2)$是待定系数

基本假设:

- 价格与销量成线性关系

利润既取决于销量和价格，也依赖于产量和成本。按照市场规律，甲的价格 p_1会随其销量 x_1的增长而降低，同时乙的销量 x_2 的增长也
会使甲的价格有稍微的下降，可以简单地假设价格与销量成线性关系，即:

$$p_1 = b_1 - a_{11}x_1 - a_{12}x_2 \qquad b_1, a_{11}, a_{12} > 0且a_{11} > a_{12}$$

同理

$$p_2 = b_2 - a_{21}x_1 - a_{22}x_2 \qquad b_2, a_{21}, a_{22} > 0$$

- 成本与产量成负指数关系

甲的成本随其产量的增长而降低，且有一个渐进值，可以假设为负指数关系，总利润为:

$$z(x_1, x_2) = (p_1 - q_1)x_1 + (p_2 - q_2)x_2$$

若根据大量的统计数据，求出系数
$b_1 = 100, a_{11} = 1, a_{12} = 0.1, b_2 = 280, a_{21} = 0.2, a_{22} = 2, r_1 = 30, \lambda_1 = 0.015, c_1 = 20, r_2 = 100, \lambda_2 = 0.02, c_2 = 30$
，则问题转化为无约束优化问题：求甲，乙两个牌号的产量 x_1，x_2，使总利润 z 最大。为简化模型，先忽略成本，并令 $a_{12} = 0, a_{21} = 0$，问题转化为求:

$$z_1 = (b_1 - a_{11}x_1)x_1 + (b_2 - a_{22}x_2)x_2$$

的极值.。显然其解为 $x_1=\frac{b_1}{2a_{11}}=50, x_2=\frac{b_2}{2a_{22}}=70$，把它作为原问题的初始值。

模型求解：

建立文件fun.m：

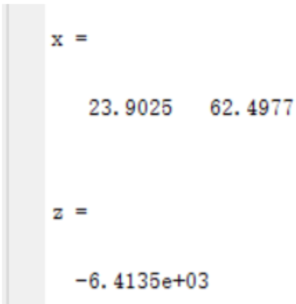
```
function f = fun(x)
y1=((100-x(1)- 0.1*x(2))-(30*exp(-0.015*x(1))+20))*x(1);
y2=((280-0.2*x(1)- 2*x(2))-(100*exp(-0.02*x(2))+30))*x(2);
f=-y1-y2;
```

输入命令：

```
x0=[50,70];
x=fminunc(‘fun’,x0),
z=fun(x)
```

计算结果：

$$x=23.9025,62.4977 \qquad z=6.4135e+003$$



```
x =

   23.9025   62.4977


z =

  -6.4135e+03
```

即甲的产量为23.9025，乙的产量为62.4977，最大利润为6413.5。