西南交通大学运筹学实验报告

作　　者：王倩妮

学　　号：2015112956

班　　级：交通 2015-02 班

任课教师：寇玮华

2017 年 6 月 10 日

西南交通大学运筹学 Lingo 实验报告

班级：交通 2015-02 班　　姓名：王倩妮　学号：2015112956

实验地点：犀浦 X5511　实验日期：2017 年 5 月、6 月每周三

一、实验目的要求：

本学期的运筹学实验，通过学习 LINGO 软件对于运筹学问题进行求解，学习 LINGO 软件的编码规则，加深了我们对于运筹学课程所学内容的理解，同时对于今后的学习与研究有着重要的作用。

LINGO 全称是 Linear interactive and General Optimizer 的缩写，是交互式的线性和通用优化求解器。LINGO 是可以帮助我们快速、方便和有效的构建和求解线性、非线性和整数最优化模型的功能全面的工具。包括功能强大的建模语言，建立和编辑问题的全功能环境，读取和写入 Excel 和数据库的功能，和一系列完全内置的求解程序。其简介的界面设计与较快的运算速度深受我们的喜爱。

通过之前的使用，不难发现，LINGO 具有的优点很多，主要有以下的几个方面：

1. 简单的模型表示
2. 方便的数据输入和输出选择
3. 强大的求解器
4. 交互式模型或创建 Turn-key 应用程序
5. 广泛的文件和 HELP 功能

通过 LINGO 的学习，一方面可以帮助我们解决之前用笔难以计算出的问题，简化计算步骤、减少求解时间，可以帮助我们解决更多复杂的问题。另一方面，通过接触另一种编程语言，可以提高我们的编程能力，适应未来的学习与发展。

LINGO 等软件的学习不仅需要依靠实验课程，还需要在课下积极实践，在应用中加深体会与熟练度，通过在今后数学建模及交通方向的研究之中的使用，加深理解。

二、实验内容：

1. LINGO 快速入门
2. LINGO 中的集

为什么使用集、什么是集、模型的集部分、定义原始集、定义派生集

3. 模型的数据部分和初始部分

模型的数据部分（基本概念、参数、实时数据处理、指定属性为一个值、数据部分的未知数值）

模型的初始部分

4. LINGO 函数

基本运算符、数学函数、金融函数、概率函数、变量界定函数、集操作

函数、集循环函数、输入输出函数、辅助函数

5. LINGO WINDOWS 命令

文件菜单、编辑菜单、LINGO 菜单、窗口菜单、帮助菜单

6. LINGO 命令行命令

7. 综合实例练习：

求解非线性方程组、装配线平衡模型、旅行售货员问题、最短路问题、露天矿生产的车辆安排、最小生成树问题、分配问题、二次分配问题、面试问题

三、学习心得体会：

本学期，通过运筹学课程的学习与运筹学实验课程的上机实验，有很多收获。记得第一次使用 LINGO 是在大一下学期学习数学建模课程期间，入门 LINGO 进行问题的求解，由于对于软件语法的不熟悉，最后没能求解成功，转而用 MATLAB 进行求解。之后的大二，在数学建模比赛之中遇到优化问题，开始渐渐上手 LINGO 软件，摸索出其语法规律，可以按照别人的程序针对自己需要求解的问题进行程序的修改。而本学期的运筹学实验，是一次较为系统的 LINGO 学习，我们也在学习之中发现了利用 LINGO 求解问题的简便性与编程语言的友好性。其界面虽然十分简洁，但并没有因为时间的流逝而被淘汰，说明其应用还是有其他软件无法比拟的优势的。

与 MATLAB 相比，LINGO 的编程语言更加简洁易懂，相信在今后的交通方向的学习之中，会有利用 LINGO 进行计算的机会。相信有了这几次的上机操作，以后也会更加容易上手操作。

四、实验中遇到的问题：

在 LINGO 实验之中，我遇到了一些问题，主要有以下几点问题：

1. 代码输入的正确性问题，代码的语法错误会影响运行结果的正确性。比如代码中英文标点所带来的问题。
2. 在运用 LINGO 求解问题时，进行文档（TXT 或 XLS）的导入会出现问题。
3. 由于不同版本的 LINGO 由维数的限制，在求解大型规划问题的时候会出现维数超界，无法求解的问题。
4. 对于程序给出的运行结果不能完全理解，由于 LINGO 的语言与帮助均为英文，对于其运行后的结果或在程序使用过程之中对于英文的小障碍使我不能理解一些运行结果的含义。

五、实验代码及截图：

➢ 1.LINGO软件学习与使用入门

补充练习：

运筹学上篇内容主要是线性规划的相关知识，首先讲解了线性规划的基本知识，而后学习了灵敏度分析，又对运输问题、指派问题、整数规划、动态规划等问题做了阐述。因此在补充练习部分，增添了运筹学上篇内容的几个例题，以加深对这方面知识的理解。针对运筹学下篇内容的补充练习在本实验报告的第七部分。

例：简单线性规划

$$\max \quad z = 6x_1 + 2x_2$$

$$\text{s.t.}\begin{cases} 2x_1 + x_2 \le 10 \\ x_1 + x_2 \le 8 \\ x_2 \le 7 \\ x_1, x_2 \ge 0 \end{cases}$$

```
model:
  max=6*x1+2*x2;
  2*x1+x2<10;
  x1+x2<8;
  x2<7;
end
```

例：整数规划求解

$$\text{Max} \quad z = 42x_1 + 95x_2$$

$$\begin{cases} 9x_1 + 7x_2 \le 56 \\ 7x_1 + 20x_2 \le 70 \\ x_1, x_2 \ge 0 \end{cases}$$

```
model:
  max=42*x1+95*x2;
  9*x1+7*x2<56;
  7*x1+20*x2<70;
  @gin(x1);@gin(x2);
End
```

例：（运输问题）已知某企业有甲、乙、丙三个分厂生产一种产品，其产量分别为7、9、7个单位，需运往A、B、C、D四个门市部，各门市部需要量分别为3、5、7、8个单位。已知单位运价如下表。试确定运输计划使总运费最少。

	A	B	C	D
甲	12	13	10	11

乙	10	12	14	10
丙	14	11	15	12

问题分析：这个问题是产销平衡问题，设 $x_{ij}(i=1,2,3;\ j=1,2,3,4)$ 代表从第 i 个产地运往第 j 个销地的数量，z 为总运费。a_i 表示第 i 个产地的产量，b_j 表示第 j 个销地的销量 c_{ij} 表示从第 i 个产地运往第 j 个销地的单位产品运输费用。所以建立了如下所示模型。

$$\max Z=\sum_{i=1}^{3}\sum_{j=1}^{4}c_{ij}x_{ij}$$

$$\begin{cases}\sum_{j=1}^{4}x_{ij}=a_i\\ \sum_{i=1}^{3}x_{ij}=b_j\\ x_{ij}\geq 0, i=1,2,3;\ j=1,2,3,4\end{cases}$$

```
model:
sets:
warehouses/wh1..wh3/: capacity;
vendors/v1..v4/: demand;
links(warehouses,vendors): cost, volume;
endsets
data:
  capacity=7 9 7;
  demand=3 5 7 8;
  cost= 12 13 10 11
        10 12 14 10
        14 11 15 12;
enddata
min=@sum(links(I,J): cost(I,J)*volume(I,J));
@for(vendors(J):
    @sum(warehouses(I): volume(I,J))=demand(J));
@for(warehouses(I):
    @sum(vendors(J): volume(I,J))<=capacity(I));
end
```

求解得到的最优调运方案为：甲→C：7单位；甲→D：0单位；乙→A：3单位；乙→D：6单位；丙→B：5单位；丙→D：2单位。计算所得最少总运费为：239。

例：（指派问题）现在要在五个工人中确定四个人来分别完成四项工作中的一项工作。由于每个工人的技术特长不同，他们完成各项工作所需的工时也不同。每个工人完成各项工作所需工时如下表所示，试找出一个工作分配方案，使总

工时最小。

工人＼工作	A	B	C	D
Ⅰ	9	4	3	7
Ⅱ	4	6	5	6
Ⅲ	5	4	7	5
Ⅳ	7	5	2	3
Ⅴ	10	6	7	4

$$\text{设 } x_{ij}=\begin{cases}1, & \text{当第}i\text{个人完成某}j\text{项工作}\\ 0, & \text{当第}i\text{个人不完成某}j\text{项工作}\end{cases}$$

$$\min Z=\sum_{i=1}^{5}\sum_{j=1}^{4}c_{ij}x_{ij}$$

$$\begin{cases}\sum_{i=1}^{5}x_{ij}=1, & j=1,2,3,4\\ \sum_{j=1}^{4}x_{ij}=1, & i=1,2,3,4,5\\ x_{ij}=0\text{或}1, & i=1,2,3,4,5;\ j=1,2,3,4\end{cases}$$

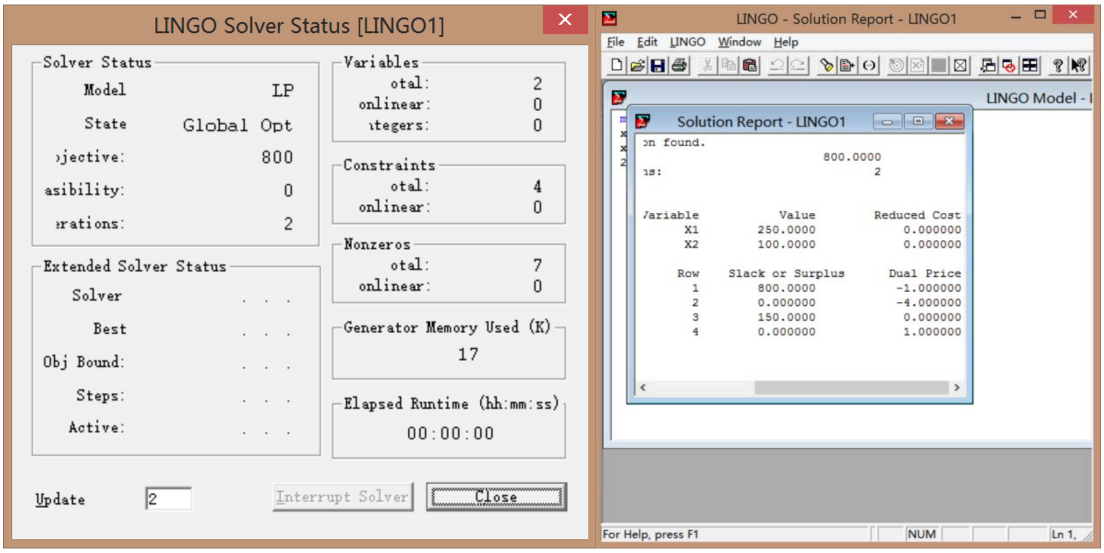
```
model:
sets:
  workers/w1..w5/;
  jobs/j1..j4/;
  links(workers,jobs): cost,volume;
Endsets
data:
  cost=9 4 3 7
       4 6 5 6
       5 4 7 5
       7 5 2 3
       10 6 7 4;
enddata
min=@sum(links: cost*volume);
  @for(workers(I): @sum(jobs(J): volume(I,J))<=1);
  @for(jobs(J): @sum(workers(I): volume(I,J))=1);
  @for(links(i,j): @bin(volume(i,j)));
End
```

对于以上问题求解，最优指派方案为：Ⅰ→C；Ⅱ→A；Ⅲ→B；Ⅳ→D。最小总工时为14工时。

PDF练习：

例 1.1

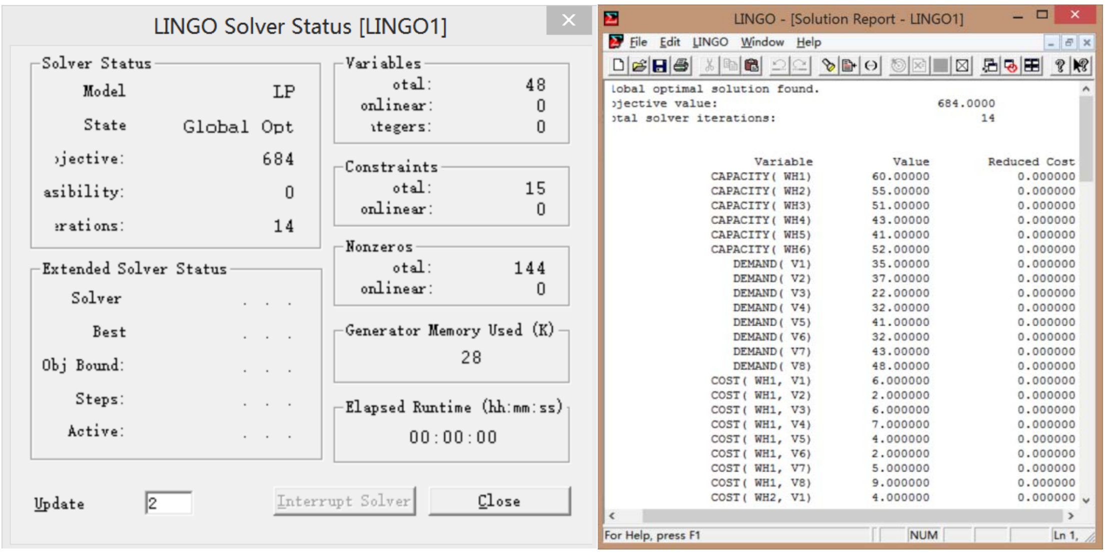
要点：利用 LINGO 解决 LP 问题




```
min=2*x1+3*x2;
x1+x2>=350;
x1>=100;
2*x1+x2<=600;
```

例 1.2

要点：利用 LINGO 解决运输问题




```
model:
sets:
  warehouses/wh1..wh6/:capacity;
  vendors/v1..v8/:demand;
  links(warehouses,vendors):cost,volume;
```

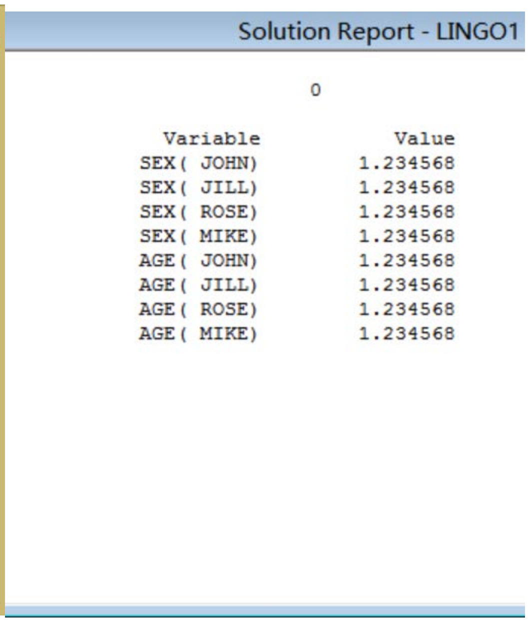
```
endsets
  min=@sum(links:cost*volume);
  @for(vendors(J):
    @sum(warehouses(I):volume(I,J))=demand(J));
  @for(warehouses(I):
    @sum(vendors(J):volume(I,J))<=capacity(I));
data:
  capacity=60 55 51 43 41 52;
  demand=35 37 22 32 41 32 43 48;
  cost=6 2 6 7 4 2 5 9
       4 9 5 3 8 5 8 2
       5 2 1 9 7 4 3 3
       7 6 7 3 9 2 7 1
       2 3 9 5 7 2 6 5
       5 5 2 2 8 1 4 3;
enddata
end
```

感想：通过本部分的练习与学习，对于LINGO的大致工作原理、编写规则、结果窗口中的显示形式有了了解，理解了程序运行结果的含义。

➢ 2.LINGO中的集

例 2.1

要点：LINGO 定义集

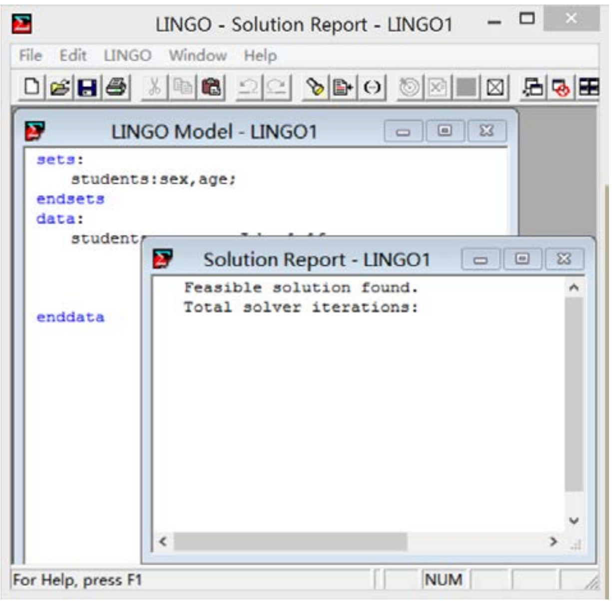



```
sets:
    students/John Jill, Rose Mike/: sex, age;
```

```
endsets
```

例 2.2

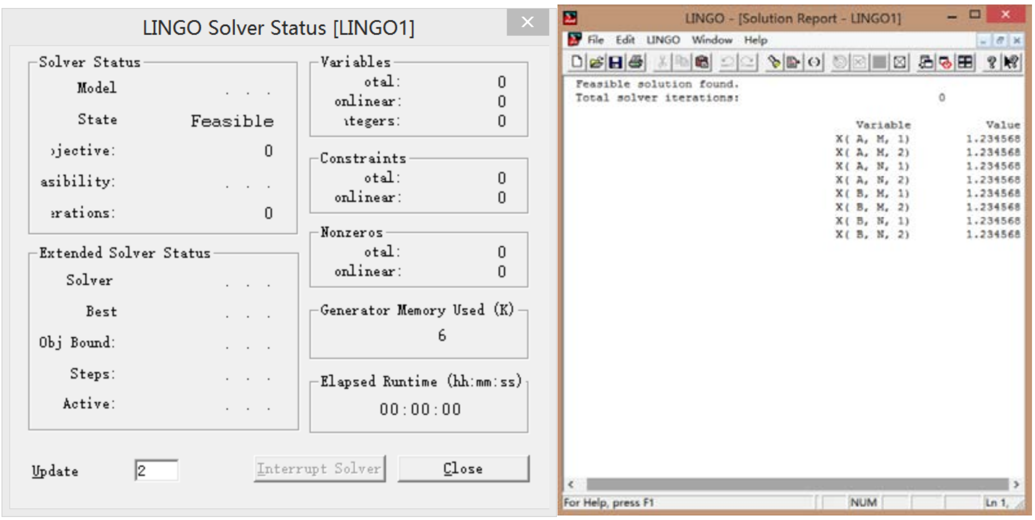
要点：集成员在数据部分定义




```
sets:
    students:sex,age;
endsets
data:
    students,sex,age=John 1 16
                     Jill 0 14
                     Rose 0 17
                     Mike 1 13;
Enddata
```

例 2.3

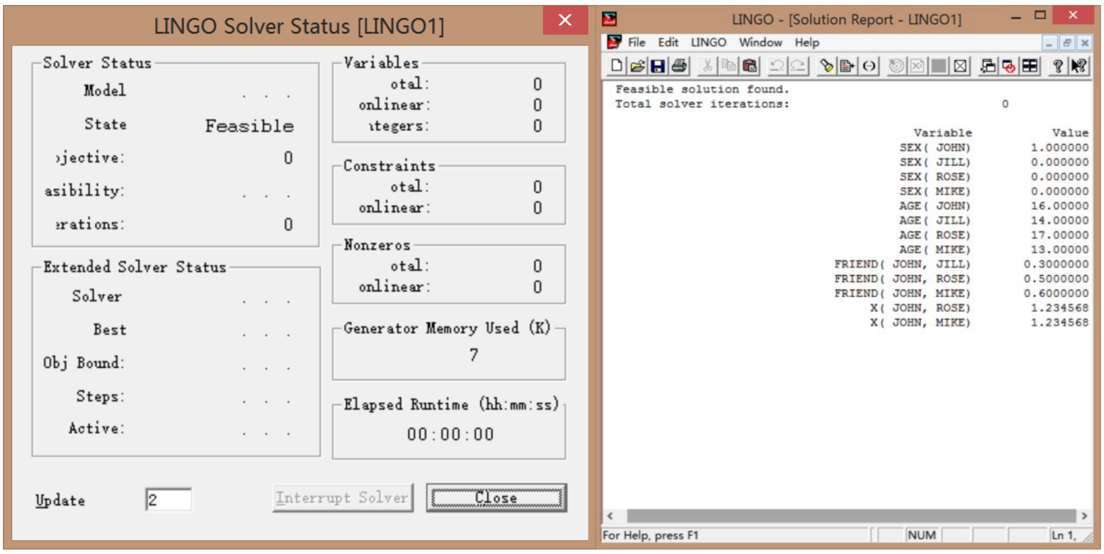
要点：LINGO 定义派生集




```
sets:
  product/A B/;
  machine/M N/;
  week/1..2/;
  allowed(product,machine,week):x;
endsets
```

例 2.4

要点：派生集过滤逻辑条件为假的成员




```
sets:
  students/John,Jill,Rose,Mike/:sex,age;
  linkmf(students,students)|sex(&1)#eq#1#and#sex(&2)#eq#0:friend;
  linkmf2(linkmf)|friend(&1,&2)#ge#0.5:x;
endsets
```

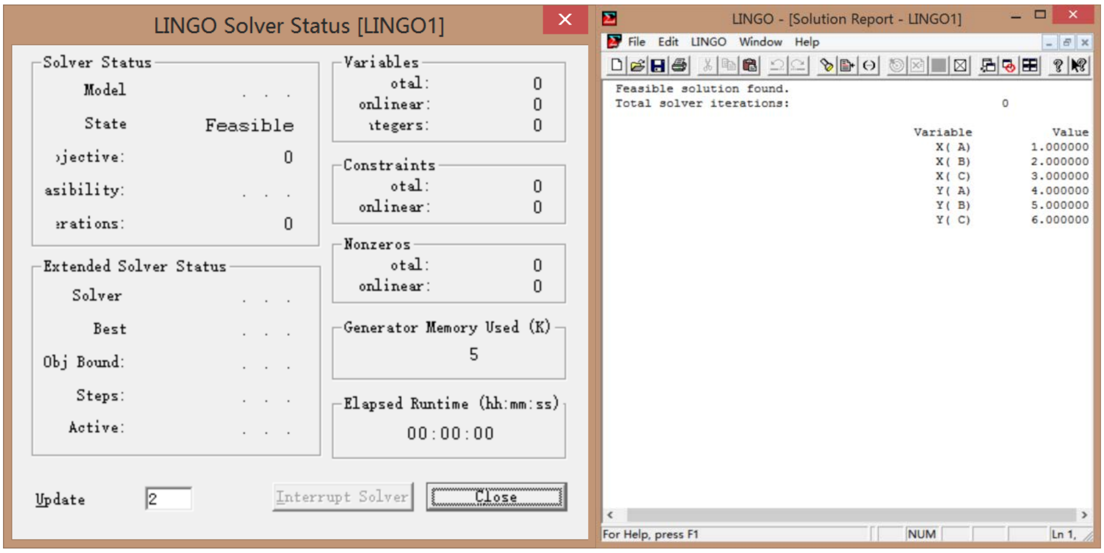
```
data:
  sex,age=1 16
          0 14
          0 17
          0 13;
  friend=0.3 0.5 0.6;
enddata
```

感想：通过本部分的练习与学习，对于LINGO中集的概念有了理解，知道了如何去定义一个集，同时也在实践的过程中发现了集定义语法错误的状态下产生的结果，在对于代码进行修正的过程中，加深了对于知识的理解。

➢ 3.模型的数据部分和初始部分

例 3.1

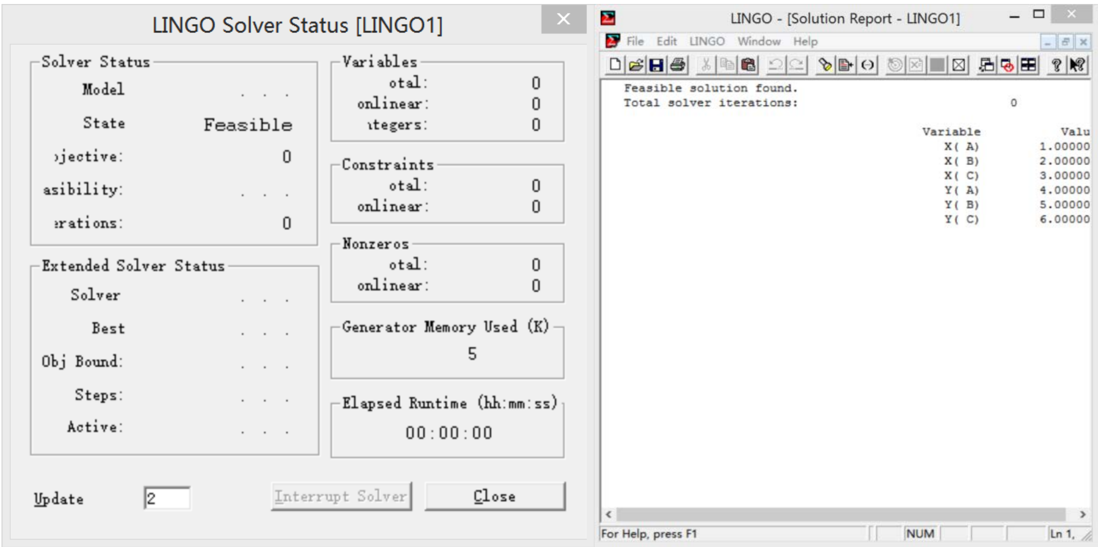
要点：LINGO 中的数据声明




```
sets:
  set1/A,B,C/:X,Y;
endsets
data:
  X=1,2,3;
  Y=4,5,6;
Enddata
```

例 3.2

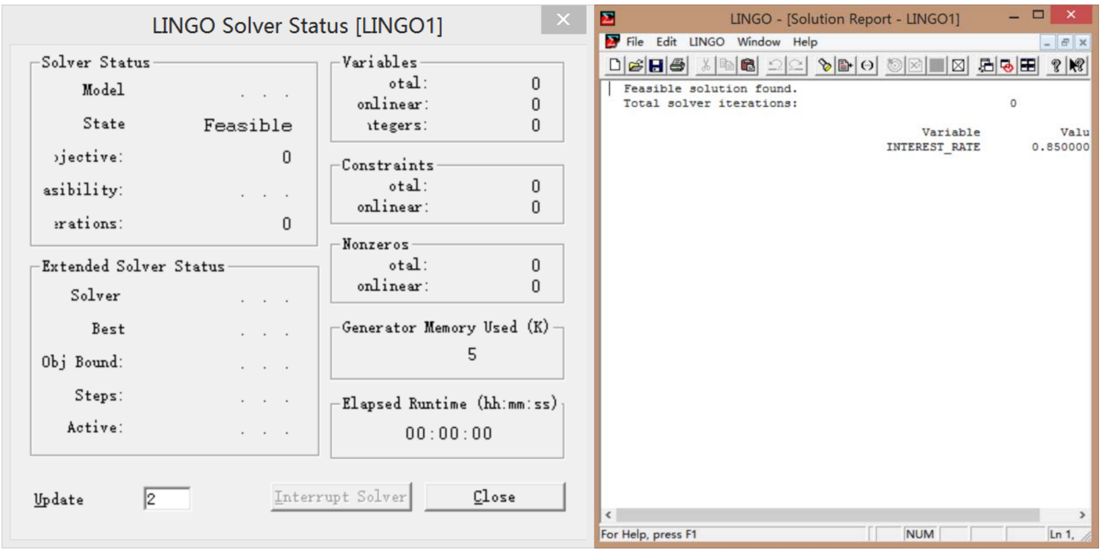
要点：LINGO 中的数据声明




```
sets:
  set1/A,B,C/:X,Y;
endsets
data:
  X,Y=1 4
      2 5
      3 6;
Enddata
```

例 3.3

要点：LINGO 中的参数设定

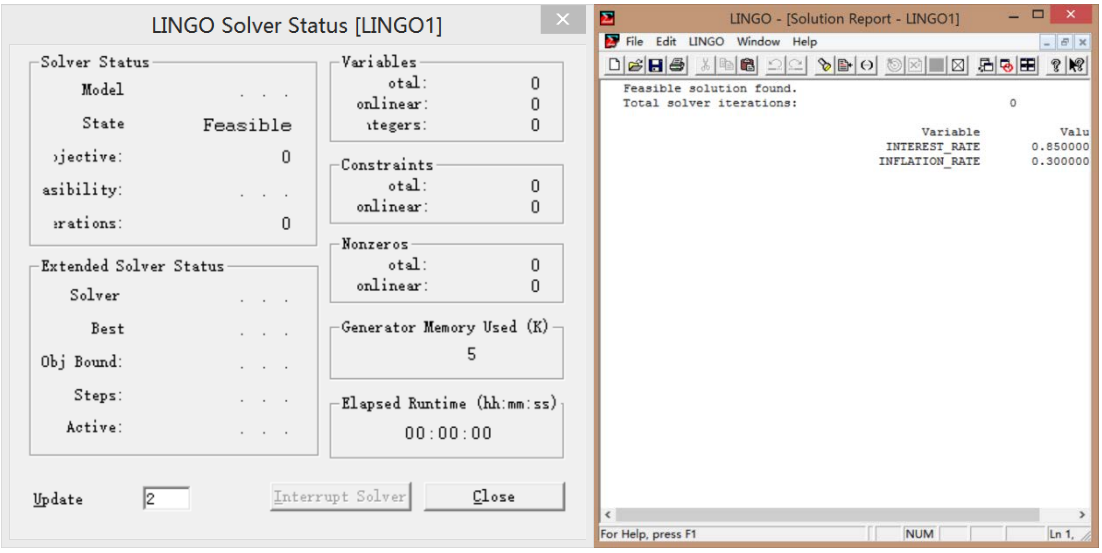



```
data:
```

```
  interest_rate=.085;
enddata
```

例 3.4

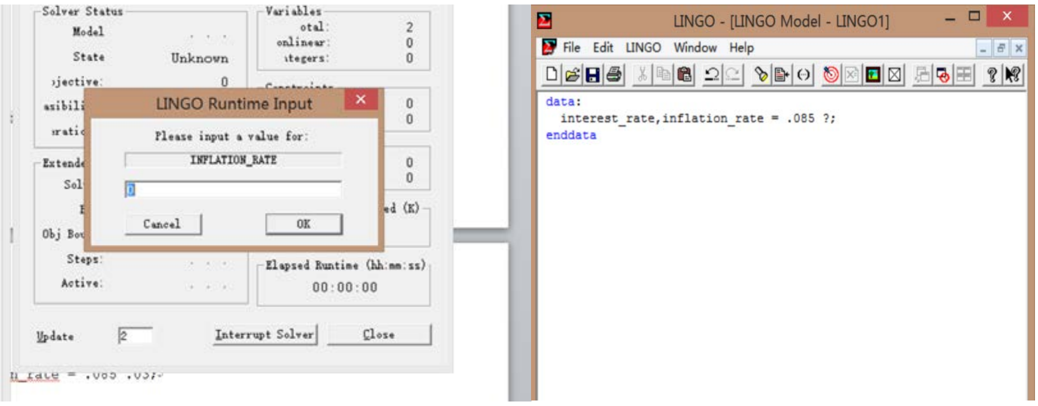
要点：LINGO 中的参数设定




```
data:
  interest_rate,inflation_rate = .085 .03;
enddata
```

例 3.5

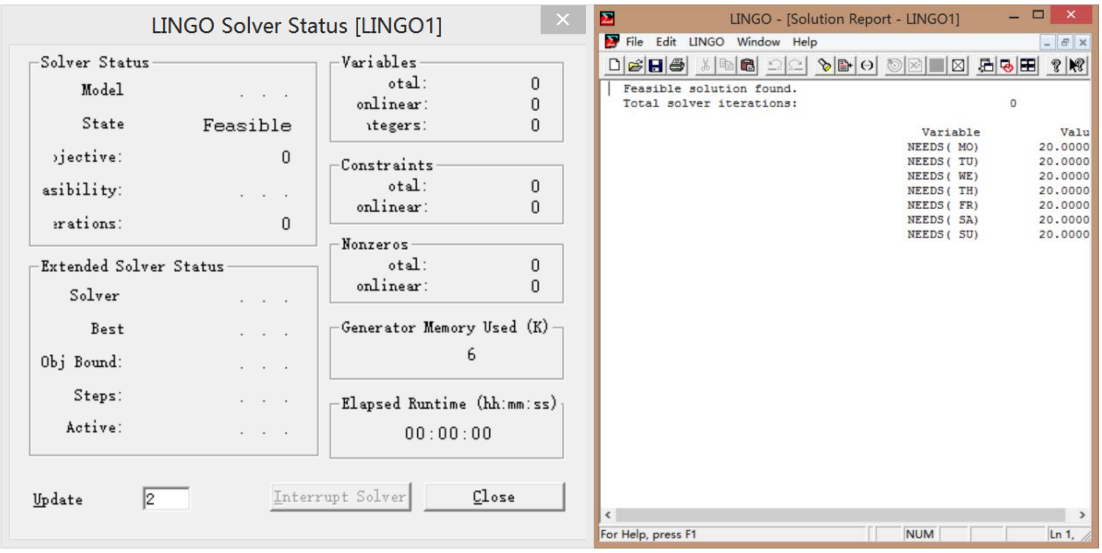
要点：LINGO 中实时数据的处理




```
data:
  interest_rate,inflation_rate = .085 ?;
enddata
```

例 3.6

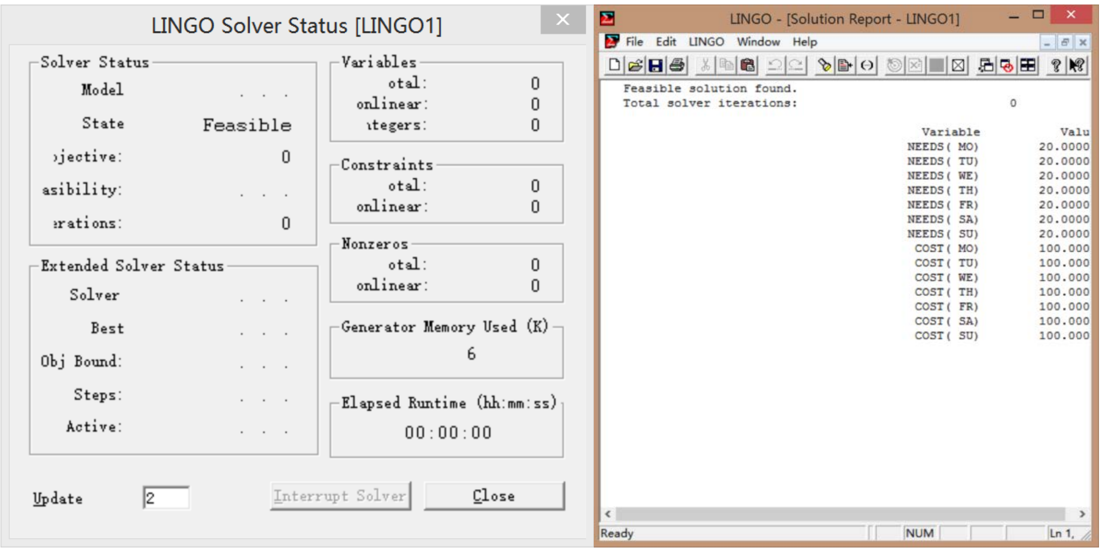
要点：LINGO 中指定属性为一个值




```
sets:
  days /MO,TU,WE,TH,FR,SA,SU/:needs;
endsets
data:
  needs=20;
enddata
```

例 3.7

要点：LINGO 中指定多个属性的值

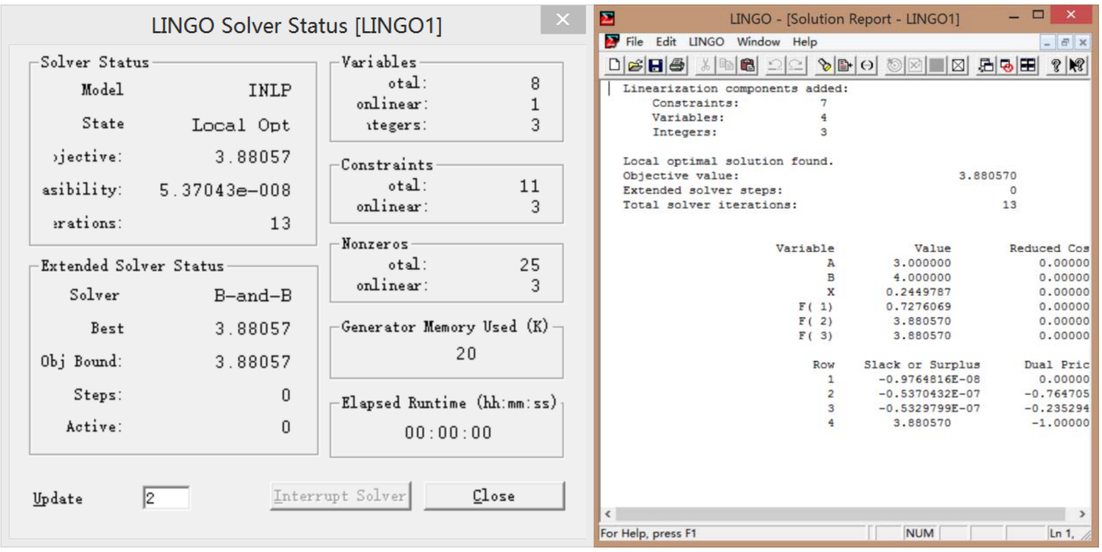



```
sets:
  days /MO,TU,WE,TH,FR,SA,SU/:needs,cost;
endsets
```

```
data:
  needs cost=20 100;
enddata
```

例 3.8

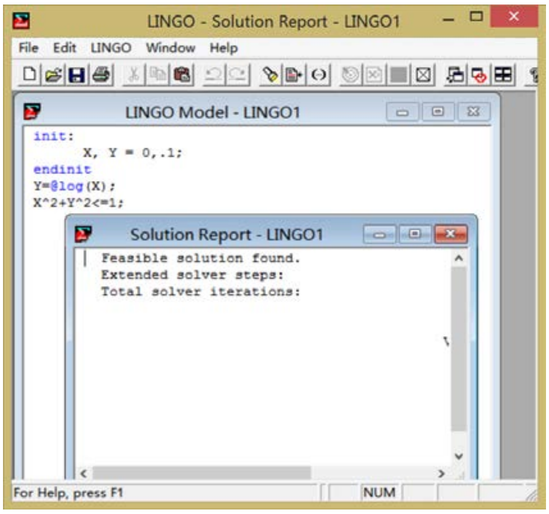
要点：LINGO 中数据部分的未知数值



```
model:
sets:
  object/1..3/:f;
endsets
data:
  a,b=3,4;
enddata
  f(1)=a*@sin(x);
  f(2)=b*@cos(x);
  f(3)=a*@cos(x)+b*@sin(x);
  min=@smax(f(1),f(2),f(3));
  @bnd(0,x,1.57);
End
```

例 3.9

要点：LINGO 模型初始部分的开始与结束



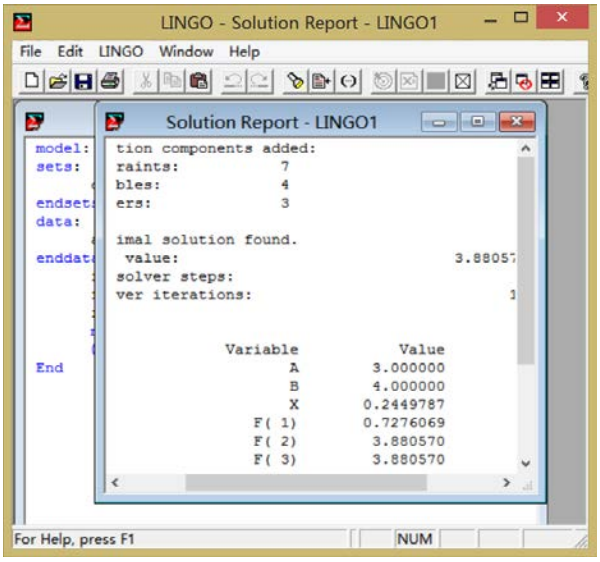

```
init:
    X, Y = 0,.1;
endinit
Y=@log(X);
X^2+Y^2<=1;
```

感想：通过本部分的学习，我对于LINGO程序中数据的定义、处理有了更加深刻的认识。学会了如何进行参数的设置，也结合前面所学的“集”的概念，便于这部分内容的学习、深入。

➢ 4.LINGO函数

例 4.3

要点：学习 LINGO 中函数的使用方法，以三角形为例

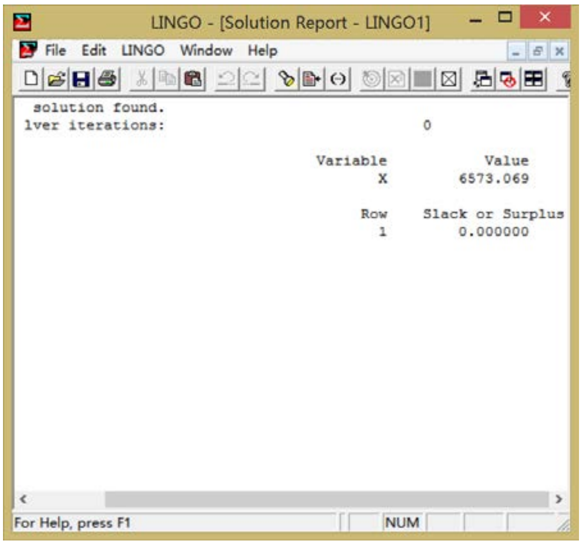



```
model:
```

```
sets:
    object/1..3/: f;
endsets
data:
    a,b = 3,4;
enddata
    f(1)= a*@sin(x);
    f(2)= b*@cos(x);
    f(3)= a*@cos(x)+b*@sin(x);
    min = @smax(f(1),f(2),f(3));
    @bnd(0,x,1.57);
End
```

例 4.4

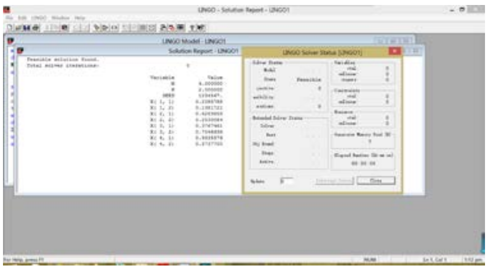
要点：LINGO 中的金融函数




```
50000 = x * @fpa(.0531,10);
```

例 4.5

要点：LINGO 中的随机数产生函数@qrand(seed)

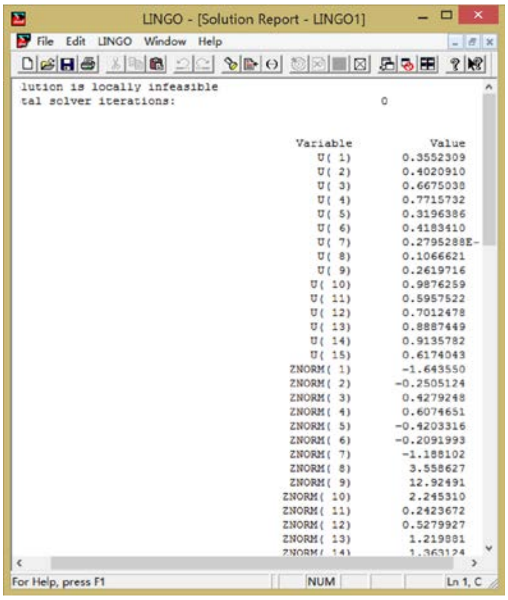


```
model:
```

```
data:
    M=4; N=2; seed=1234567;
enddata
sets:
    rows/1..M/;
    cols/1..N/;
    table(rows,cols): x;
endsets
data:
    X=@qrand(seed);
enddata
end
```

例 4.6

要点：不指定种子，LINGO 利用系统时间构造种子




```
model:
sets:
    series/1..15/: u, znorm, zt;
endsets
    u( 1) = @rand( .1234);
    @for(series( I)| I #GT# 1:
        u( I) = @rand( u( I - 1))
    );
    @for( series( I):
        @psn( znorm( I)) = u( I);
        @ptd( 2, zt( I)) = u( I);
        @free( znorm( I)); @free( zt( I));
```

```
    );
End
```

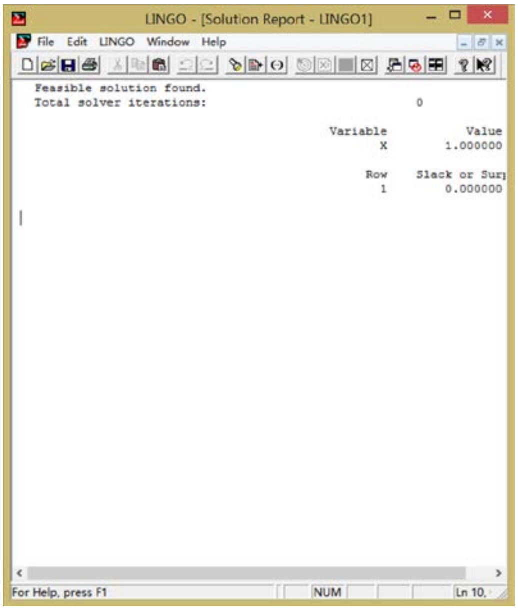
例 4.7（结果为 0）

要点：LINGO 中集函数的操作

```
sets:
    I/x1..x4/;
    B(I)/x2/;
    C(I)|#not#@in(B,&1):;
endsets
```

例 4.8

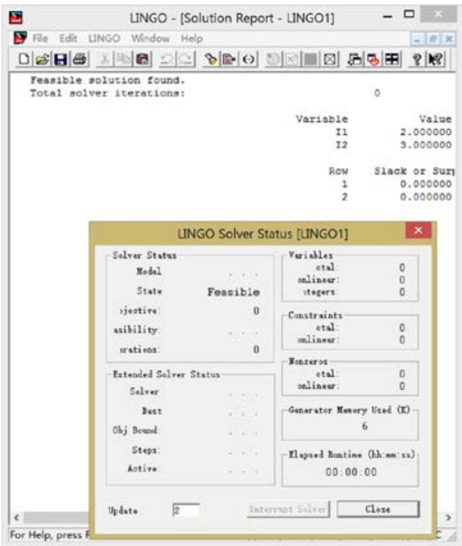
要点：确定集函数元素属于派生集




```
sets:
    S1/A B C/;
    S2/X Y Z/;
    S3(S1,S2)/A X, A Z, B Y, C X/;
endsets
    X=@in(S3,@index(S1,B),@index(S2,Y));
```

例 4.9

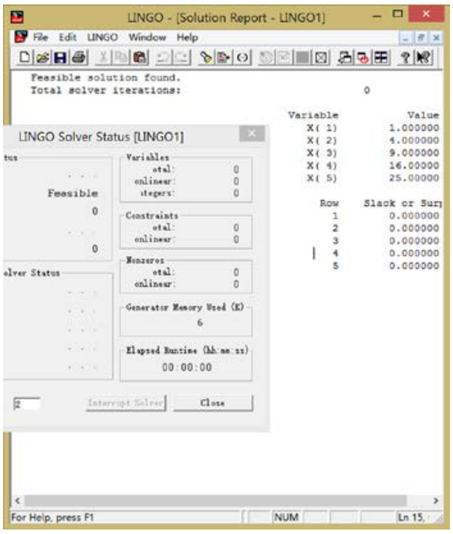
要点：@index 制定集的必要性




```
sets:
    girls/debble,sue,alice/;
    boys/bob,joe,sue,fred/;
endsets
I1=@index(sue);
I2=@index(boys,sue);
```

例 4.10

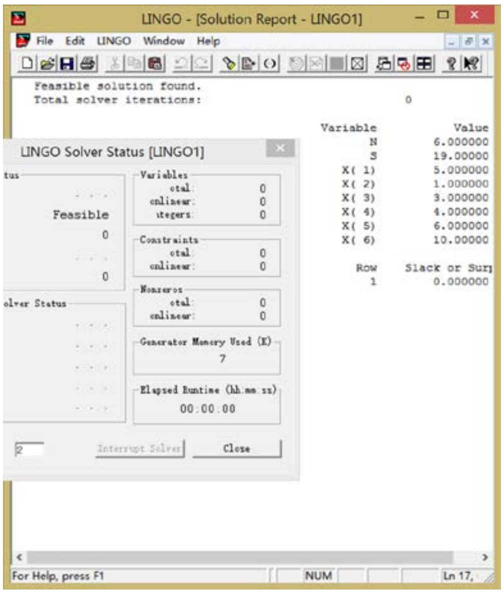
要点：LINGO 集函数循环的 for 循环




```
model:
sets:
    number/1..5/:x;
endsets
    @for(number(I): x(I)=I^2);
End
```

例 4.11

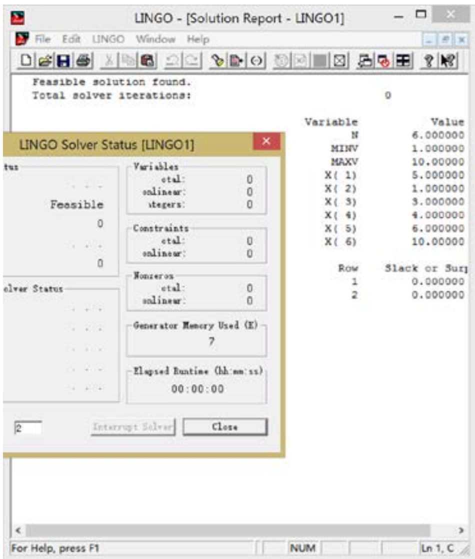
要点：集函数的@sum 函数




```
model:
data:
    N=6;
enddata
sets:
    number/1..N/:x;
endsets
data:
    x = 5 1 3 4 6 10;
enddata
    s=@sum(number(I) | I #le# 5: x);
end
```

例 4.12

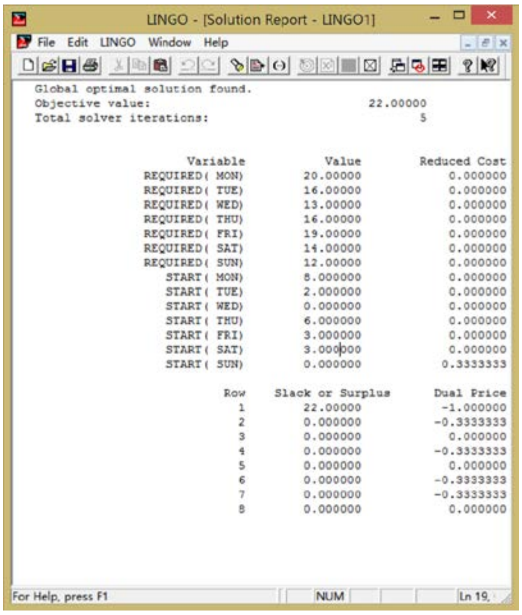
要点：集函数的@min，@max 函数




```
model:
data:
    N=6;
enddata
sets:
    number/1..N/:x;
endsets
data:
    x = 5 1 3 4 6 10;
enddata
    minv=@min(number(I) | I #le# 5: x);
    maxv=@max(number(I) | I #ge# N-2: x);
end
```

例 4.13

要点：职员时序安排模型




```
Global optimal solution found.
Objective value:                              22.00000
Total solver iterations:                             5

                       Variable           Value        Reduced Cost
                 REQUIRED( MON)        20.00000            0.000000
                 REQUIRED( TUE)        16.00000            0.000000
                 REQUIRED( WED)        13.00000            0.000000
                 REQUIRED( THU)        16.00000            0.000000
                 REQUIRED( FRI)        19.00000            0.000000
                 REQUIRED( SAT)        14.00000            0.000000
                 REQUIRED( SUN)        12.00000            0.000000
                    START( MON)        8.000000            0.000000
                    START( TUE)        2.000000            0.000000
                    START( WED)        0.000000            0.000000
                    START( THU)        6.000000            0.000000
                    START( FRI)        3.000000            0.000000
                    START( SAT)        3.000000            0.000000
                    START( SUN)        0.000000           0.3333333

                            Row    Slack or Surplus      Dual Price
                              1        22.00000           -1.000000
                              2        0.000000          -0.3333333
                              3        0.000000            0.000000
                              4        0.000000          -0.3333333
                              5        0.000000            0.000000
                              6        0.000000          -0.3333333
                              7        0.000000          -0.3333333
                              8        0.000000            0.000000
```

```
model:
sets:
    days/mon..sun/: required,start;
endsets
data:
    required = 20 16 13 16 19 14 12;
enddata
    min=@sum(days: start);
    @for(days(J):
        @sum(days(I) | I #le# 5:
            start(@wrap(J+I+2,7))) >= required(J));
end
```

例 4.14（无 txt 文档没法用）

要点：数据的输入输出，利用@file 将 LINGO 与 TXT 文件的连接

```
model:
sets:
    warehouses/ @file('1_2.txt') /: capacity;
    vendors/ @file('1_2.txt') /: demand;
    links(warehouses,vendors): cost, volume;
endsets
    min=@sum(links: cost*volume);
    @for(vendors(J):
        @sum(warehouses(I): volume(I,J))=demand(J));
    @for(warehouses(I):
        @sum(vendors(J): volume(I,J))<=capacity(I));
data:
```

```
    capacity = @file('1_2.txt') ;
    demand = @file('1_2.txt') ;
    cost = @file('1_2.txt') ;
enddata
end
```

例 4.15（需要与前面的配合使用）

要点：利用@text 将 LINGO 与 TXT 文件连接，完成数据的输入输出

```
model:
data:
    N=6;
enddata
sets:
    number/1..N/:x;
endsets
data:
    x = 5 1 3 4 6 10;
enddata
    minv=@min(number(I) | I #le# 5: x);
    maxv=@max(number(I) | I #ge# N-2: x);
end
model:
sets:
    days/mon..sun/: required,start;
endsets
data:
    required = 20 16 13 16 19 14 12;
    @text('d:\out.txt')=days '至少需要的职员数为' start;
enddata
    min=@sum(days: start);
    @for(days(J):
        @sum(days(I) | I #le# 5:
            start(@wrap(J+I+2,7))) >= required(J));
end
```

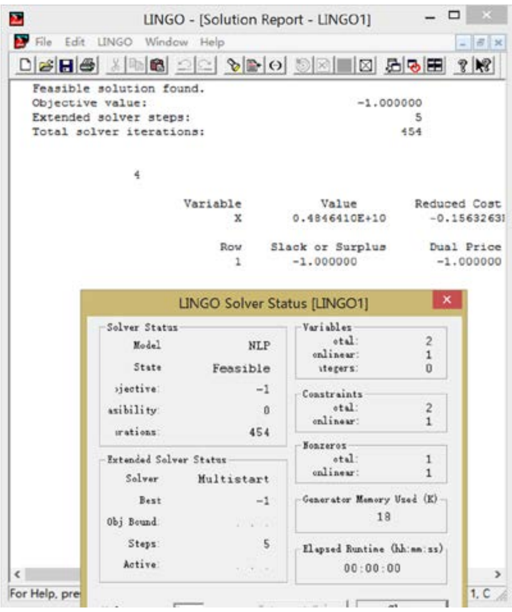
例 4.16（没有 xls 文档没法使用）

要点：利用@ole 函数使 LINGO 与 excel 连接读取数据，完成数据的输入输出

```
sets:
    PRODUCT;
    MACHINE;
    WEEK;
    ALLOWED(PRODUCT,MACHINE,WEEK):x,y;
endsets
data:
    rate=0.01;
    PRODUCT,MACHINE,WEEK,ALLOWED,x,y=@OLE('D:\IMPORT.XLS');
    @OLE('D:\IMPORT.XLS')=rate;
Enddata
```

例 4.17

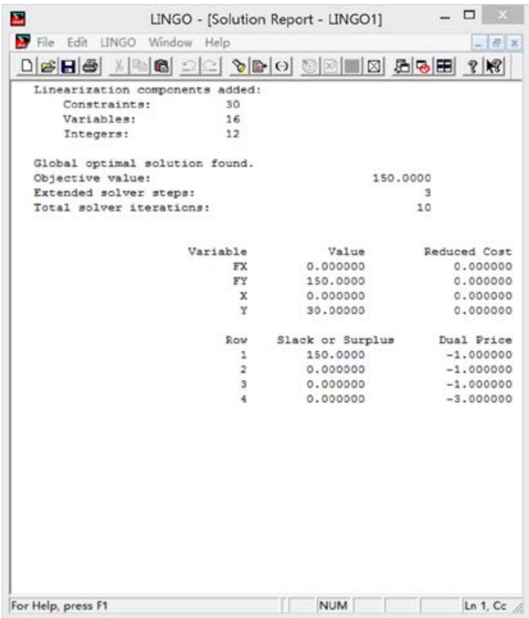
要点：@status 函数的使用方法




```
model:
min=@sin(x);
data:
    @text()=@status();
enddata
end
```

例 4.18

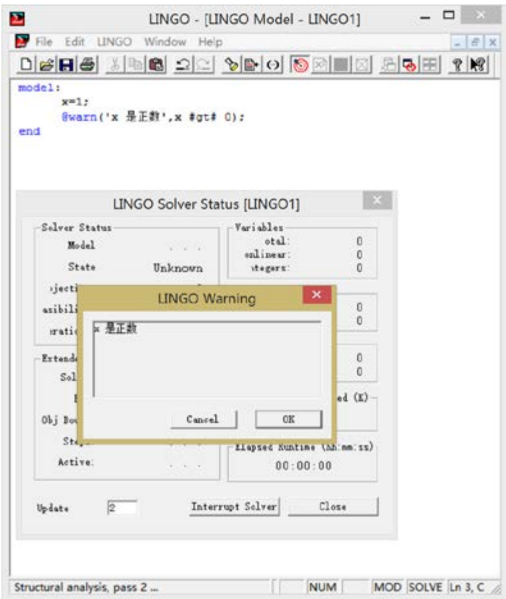
要点：辅助函数@if 函数评价逻辑表达式真假




```
model:
    min=fx+fy;
    fx=@if(x #gt# 0, 100,0)+2*x;
    fy=@if(y #gt# 0,60,-y)+3*y;
    x+y>=30;
end
```

例 4.19

要点： 利用@warn 函数进行逻辑判断




```
model:
    x=1;
    @warn('x 是正数',x #gt# 0);
```

```
End
```

感想：第四部分内容主要是关于LINGO的函数的一些应用，这些函数对于问题的解决非常有帮助，同时也应该学习借助使用“帮助”功能，解决问题。

➢ 5.LINGO Windows命令

感想：第五部分内容主要是教会我们如何理解LINGO窗口命令，从文件菜单、编辑菜单、LINGO菜单、窗口命令、帮助菜单几个部分进行了介绍，也帮助我对于LINGO给出的功能从何处设置，给出的报表反映的什么信息有了更加深刻的认识。

➢ 6.LINGO命令行命令

感想：第六部分内容主要对LINGO命令行的命令进行了介绍，包括LINGO信息命令、输入命令、显示命令、文件输出命令、求解模型命令、编辑模型命令、退出系统命令、系统参数命令等内容。

➢ 7.综合举例

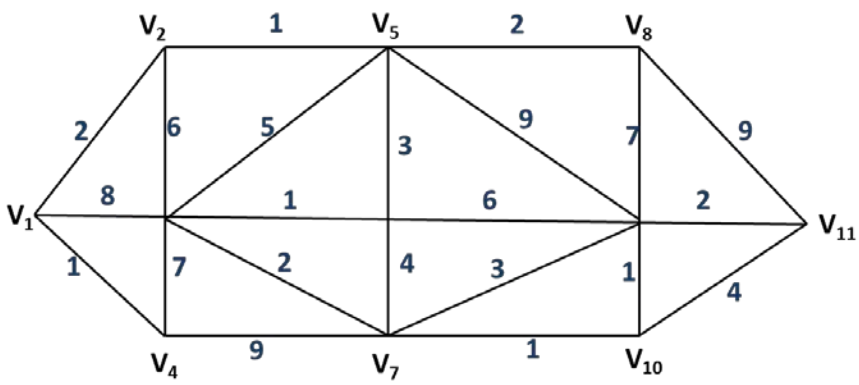
补充练习：

利用LINGO求解图与网络问题：

运筹学课程下篇内容主要是图与网络问题，因此，在实验之中，我也侧重于这方面问题的求解的联系，特此写下下面的几个具有代表性的例题。

利用LINGO，可以将图论问题通过代码写出，进而利用LINGO进行求解，在解决简单的图论问题如（最短路问题、最大流问题、最小费用最大流问题）后，还可以进一步拓展应用于解决旅行商（TSP）问题、钢管订购和运输（复杂网络）的问题。

例：（最短路）求下图无相图v_1-v_{11}的最短路



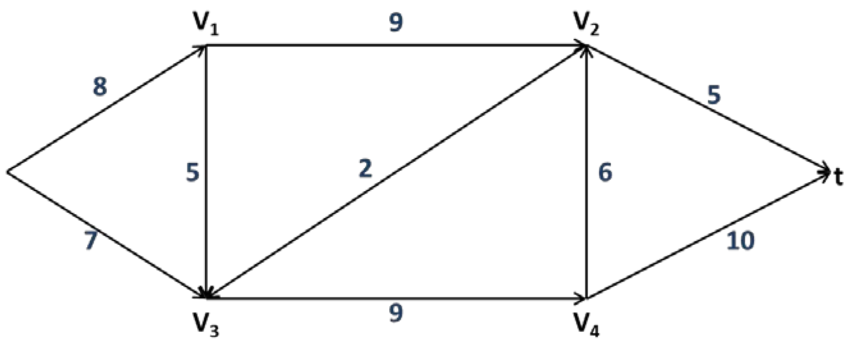

```
model:
sets:
```

```
 cities/1..11/;
roads(cities,cities):w,x;
endsets
data:
w=0;
enddata
calc:
w(1,2)=2;w(1,3)=8;w(1,4)=1;
w(2,3)=6;w(2,5)=1;
w(3,4)=7;w(3,5)=5;w(3,6)=1;w(3,7)=2;
w(4,7)=9;
w(5,6)=3;w(5,8)=2;w(5,9)=9;
w(6,7)=4;w(6,9)=6;
w(7,9)=3;w(7,10)=1;
w(8,9)=7;w(8,11)=9;
w(9,10)=1;w(9,11)=2;w(10,11)=4;
@for(roads(i,j):w(i,j)=w(i,j)+w(j,i));
@for(roads(i,j):w(i,j)=@if(w(i,j) #eq#0,1000,w(i,j)));
endcalc
n=@size(cities);
min=@sum(roads:w*x);
@for(cities(i)|i #ne#1 #and#i
#ne#n:@sum(cities(j):x(i,j))=@sum(cities(j):x(j,i)));
@sum(cities(j):x(1,j))=1;
@sum(cities(j):x(j,1))=0;
@sum(cities(j):x(j,n))=1;
@for(roads:@bin(x));
End
```

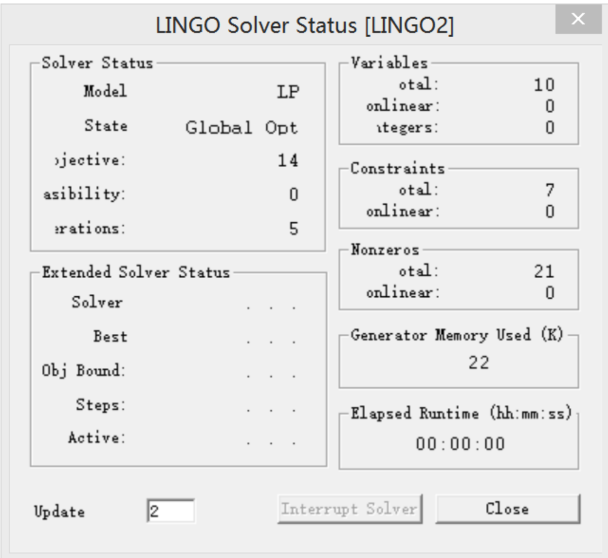
本问题的最短路径1→2→5→6→3→7→10→9→11，最短路径长为13。

若此问题为有向图则将@sum(cities(j):x(j,1))=0去掉。

例：（最大流问题）现需要将城市s的石油通过管道运送至城市t，中间有4个中转站v1,v2,v3和v4，城市与中转站的连接以及管道的容量如下图所示，求从城市s到城市t的最大流。

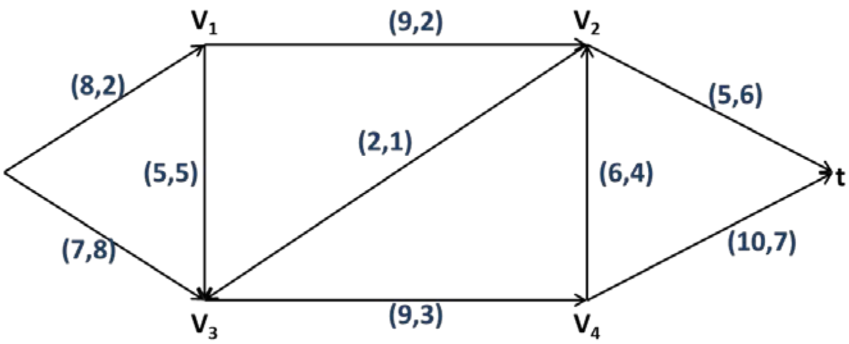


```
model:
sets:
nodes/s,1,2,3,4,t/;
arcs(nodes,nodes)/s 1,s 3,1 2,1 3,2 3,2 t,3 4,4 2,4 t/:c,f;
endsets
data:
c=8 7 9 5 2 5 9 6 10;
enddata
n=@size(nodes);
max=flow;
@for(nodes(i) |i #ne#1 #and#i #ne#n:
 @sum(arcs(i,j):f(i,j))=@sum(arcs(j,i):f(j,i)));
@sum(arcs(i,j) |i #eq# 1:f(i,j))=flow;
@sum(arcs(i,j) |j #eq# n:f(i,j))=flow;
@for(arcs:@bnd(0,f,c));
End
```

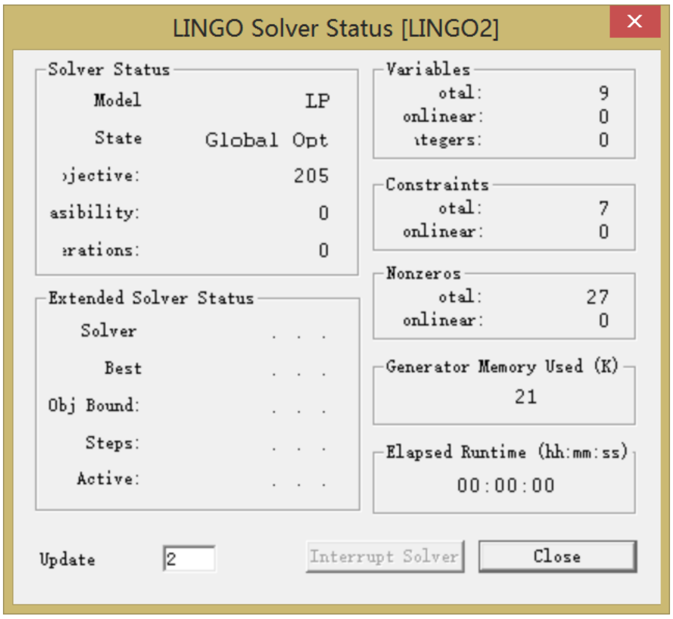


本问题得到的最大流流量为14.

例：（最小费用最大流问题）对于上一个最大流问题，由于输油管道的长短不一或地质等原因，每条管道上运输费用也不同。下图第一个数字是网络容量，第二个数字是单位运费，求最小费用最大流。



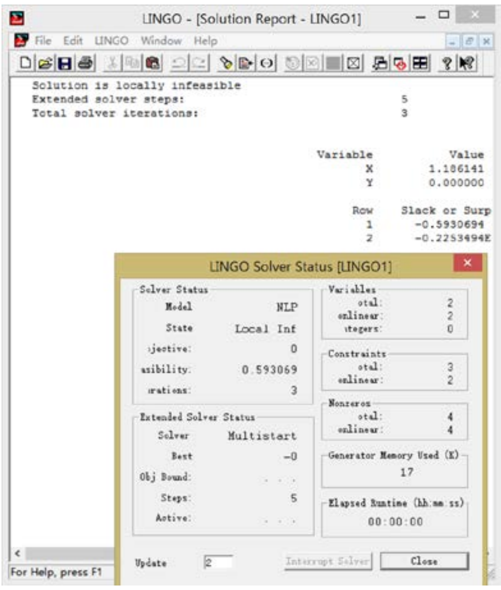
```
model:
sets:
nodes/s,1,2,3,4,t/:d;
arcs(nodes,nodes)/s 1,s 3,1 2,1 3,2 3,2 t,3 4,4 2,4 t/:b,c,f;
endsets
data:
d=14 0 0 0 0 -14;
b=2 8 2 5 1 6 3 4 7;
c=8 7 9 5 2 5 9 6 10;
enddata
min=@sum(arcs:b*f);
@for(nodes(i):@sum(arcs(i,j):f(i,j))-@sum(arcs(j,i):f(j,i))=d(i));
@for(arcs:@bnd(0,f,c));
End
```



本问题求得的最小费用最大流为205.

例 7.1

要点：利用 LINGO 求解非线性方程组




```
model:
    x^2+y^2=2;
    2*x^2+x+y^2+y=4;
End
```

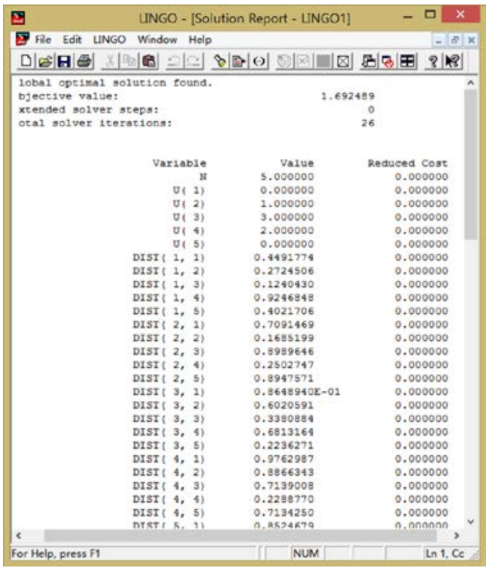
例7.2(维数超了)

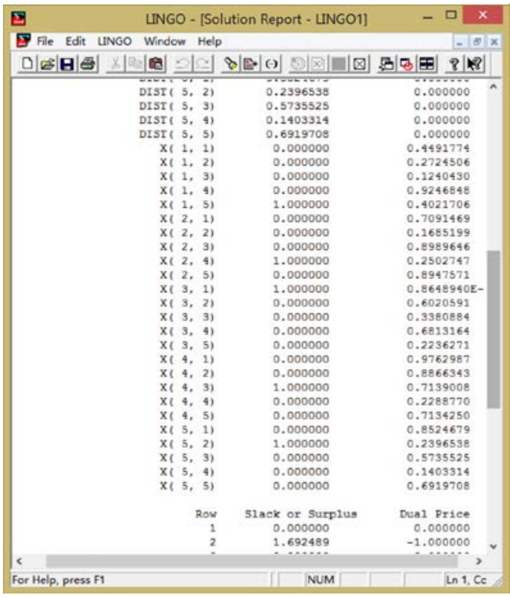
要点：利用LINGO解决图的问题——装配线的平衡模型

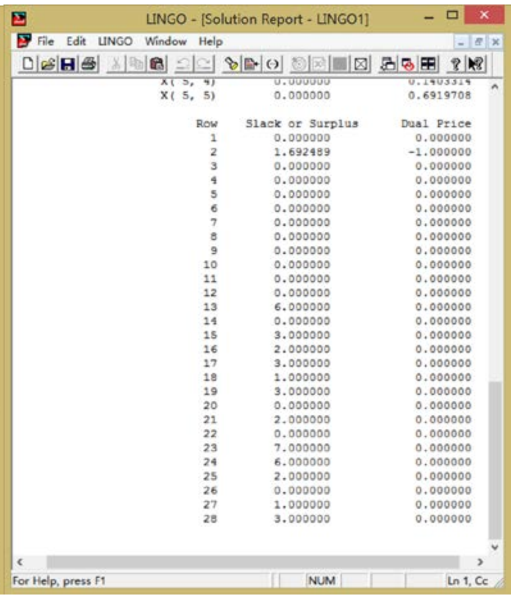
```
MODEL:
SETS:
    TASK/A B C D E F G H I J K/: T;
    PRED(TASK,TASK)/A,B B,C C,F C,G F,J G,J
     J,K D,E E,H E,I H,J I,J/;
    STATION/1..4/;
    TXS(TASK,STATION):X;
ENDSETS
DATA:
    T=45 11 9 50 15 12 12 12 12 8 9;
ENDDATA
    @FOR(TASK(I):@SUM(STATION(K):X(I,K))=1);
    @FOR(PRED(I,J):@SUM(STATION(K):K*X(J,K)-K*X(I,K))>=0);
    @FOR(STATION(K):
    @SUM(TXS(I,K):T(I)*X(I,K))<=CYCTIME);
    MIN=CYCTIME;
    @FOR(TXS:@BIN(X));
END
```

例7.3

要点：利用LINGO求解——TSP旅行商问题





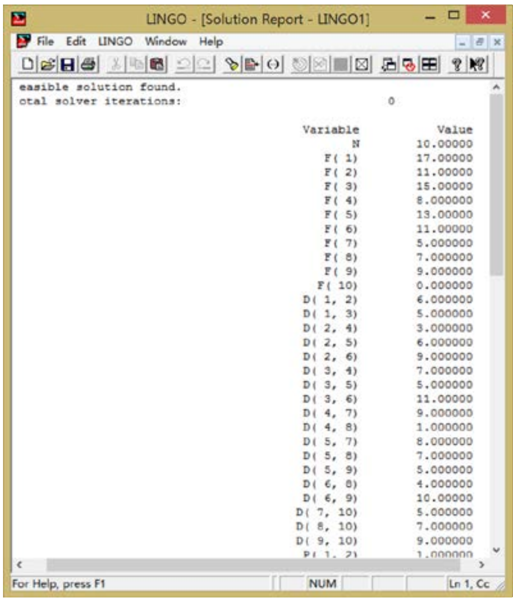



```
model:
sets:
    city/1..5/:u;
    link(city,city):
        dist,
            x;
endsets
    n=@size(city);
data:
    dist=@qrand(1);
enddata
    min=@sum(link:dist*x);
    @for(city(K):
```

```
        @sum(city(I)|I#ne#K:x(I,K))=1;
        @sum(city(J)|J#ne#K:x(K,J))=1;
    );
    @for(city(I)|I#gt#1:
        @for(city(J)|J#gt#1 #and# I #ne# J:
            u(I)-u(J)+n*x(I,J)<=n-1);
    );
    @for(city(I)|I#gt# 1:u(I)<=n-2);
    @for(link:@bin(x));
End
```

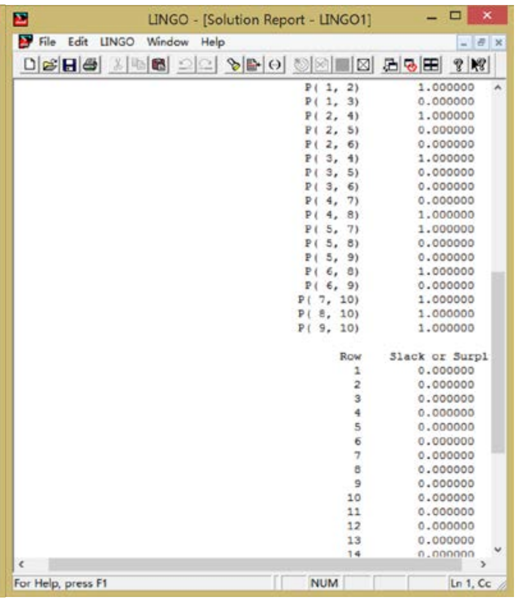
例7.4

要点：利用LINGO求解网络模型——最短路问题



```
LINGO - [Solution Report - LINGO1]
File  Edit  LINGO  Window  Help
easible solution found.
otal solver iterations:                    0

                    Variable           Value
                           N        10.00000
                       F( 1)        17.00000
                       F( 2)        11.00000
                       F( 3)        15.00000
                       F( 4)        8.000000
                       F( 5)        13.00000
                       F( 6)        11.00000
                       F( 7)        5.000000
                       F( 8)        7.000000
                       F( 9)        9.000000
                      F( 10)        0.000000
                    D( 1, 2)        6.000000
                    D( 1, 3)        5.000000
                    D( 2, 4)        3.000000
                    D( 2, 5)        6.000000
                    D( 2, 6)        9.000000
                    D( 3, 4)        7.000000
                    D( 3, 5)        5.000000
                    D( 3, 6)        11.00000
                    D( 4, 7)        9.000000
                    D( 4, 8)        1.000000
                    D( 5, 7)        8.000000
                    D( 5, 8)        7.000000
                    D( 5, 9)        5.000000
                    D( 6, 8)        4.000000
                    D( 6, 9)        10.00000
                   D( 7, 10)        5.000000
                   D( 8, 10)        7.000000
                   D( 9, 10)        9.000000
                    P( 1, 2)        1.000000
For Help, press F1        NUM        Ln 1, Cc
```



```
LINGO - [Solution Report - LINGO1]
File  Edit  LINGO  Window  Help
                    P( 1, 2)        1.000000
                    P( 1, 3)        0.000000
                    P( 2, 4)        1.000000
                    P( 2, 5)        0.000000
                    P( 2, 6)        0.000000
                    P( 3, 4)        1.000000
                    P( 3, 5)        0.000000
                    P( 3, 6)        0.000000
                    P( 4, 7)        0.000000
                    P( 4, 8)        1.000000
                    P( 5, 7)        1.000000
                    P( 5, 8)        0.000000
                    P( 5, 9)        0.000000
                    P( 6, 8)        1.000000
                    P( 6, 9)        0.000000
                   P( 7, 10)        1.000000
                   P( 8, 10)        1.000000
                   P( 9, 10)        1.000000

                         Row    Slack or Surpl
                           1        0.000000
                           2        0.000000
                           3        0.000000
                           4        0.000000
                           5        0.000000
                           6        0.000000
                           7        0.000000
                           8        0.000000
                           9        0.000000
                          10        0.000000
                          11        0.000000
                          12        0.000000
                          13        0.000000
                          14        0.000000
For Help, press F1        NUM        Ln 1, Cc
```

```
model:
data:
    n=10;
enddata
sets:
    cities/1..n/:F;
    roads(cities,cities)/
        1,2 1,3
        2,4 2,5 2,6
        3,4 3,5 3,6
        4,7 4,8
        5,7 5,8 5,9
        6,8 6 9
        7,10
```

```
        8,10
        9,10
    /:D,P;
endsets
data:
    D=
        6 5
        3 6 9
        7 5 11
        9 1
        8 7 5
        4 10
        5
        7
        9;
enddata
    F(n)=0;
    @for(cities(i)|i#lt#n:
        F(i)=@min(roads(i,j):D(i,j)+F(j));
    );
    @for(roads(i,j):
        P(i,j)=@if(F(i) #eq# D(i,j)+F(j),1,0);
    );
End
```

例7.5（提示有语法错误）

要点：数学建模03国赛露天矿问题的求解

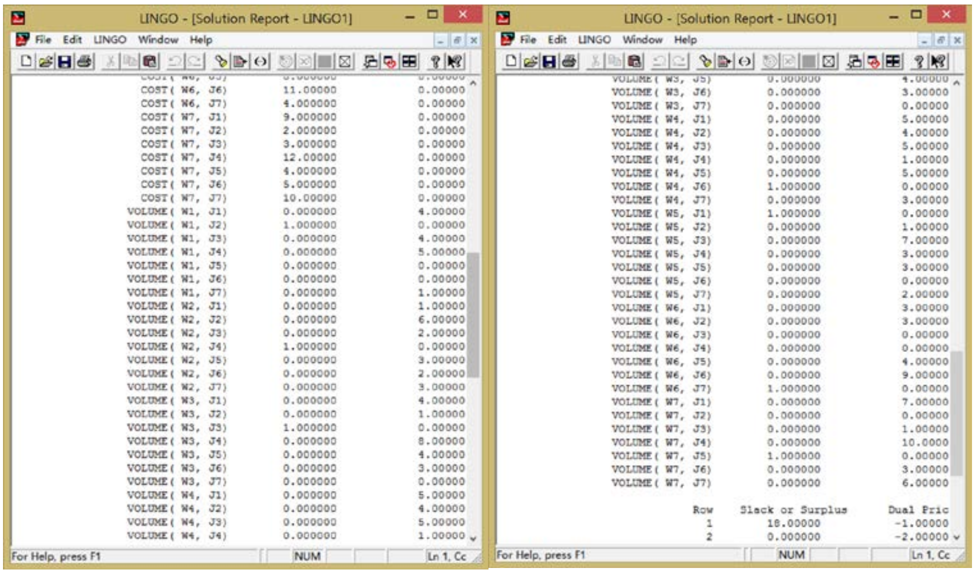
```
model:
title CUMCM-2003B-01;
sets:
cai/1..10/:crate,cnum,cy,ck,flag;
xie/1..5/:xsubject,xnum;
link(xie,cai):distance,lsubject,number,che,b;
endsets
data:
crate=30 28 29 32 31 33 32 31 33 31;
xsubject=1.2 1.3 1.3 1.9 1.3;
distance=5.26 5.19 4.21 4.00 2.95 2.74 2.46 1.90 0.64 1.27
        1.90 0.99 1.90 1.13 1.27 2.25 1.48 2.04 3.09 3.51
        5.89 5.61 5.61 4.56 3.51 3.65 2.46 2.46 1.06 0.57
        0.64 1.76 1.27 1.83 2.74 2.60 4.21 3.72 5.05 6.10
        4.42 3.86 3.72 3.16 2.25 2.81 0.78 1.62 1.27 0.50;
```

```
cy=1.25 1.10 1.35 1.05 1.15 1.35 1.05 1.15 1.35 1.25;
ck=0.95 1.05 1.00 1.05 1.10 1.25 1.05 1.30 1.35 1.25;
enddata
min=@sum(cai(i):
    @sum(xie(i):
        number(j,i)*154*distance(j,i)));
@for(link(i,j):
b(i,j)=@floor((8*60-(@floor((distance(i,j)/28*60*2+3+5)/5)-1)*5)/(dis
tance(i,j)/28*60*2+3+5)));
@for(link(i,j):
lsubject(i,j)=(@floor((distance(i,j)/28*60*2+3+5)/5))*b(i,j));
@for(cai(j):
    cnum(j)=@sum(xie(i):number(i,j)));
@for(xie(i):
    xnum(i)=@sum(cai(j):number(i,j)));
@for(link(i,j):
    number(i,j)<=lsubject(i,j));
@for(cai(j):
    cnum(j)<=flag(j)*8*60/5);
@sum(cai(j): flag(j) ) <=7;
@for(xie (i):
xnum(i)<=8*20);
@for(cai (i): number(1,i)+number(2,i)+number(5,i)<=ck(i)*10000/154);
@for(cai(i): number(3,i)+number(4,i)<=cy(i)*10000/154);
@for(xie(i):
xnum(i)>= xsubject (i)*10000/154);
@sum(cai(j):
number(1,j)*(crate(j)-30.5))<=0;
@sum(cai(j):
number(2,j)*(crate(j)-30.5))<=0;
@sum(cai(j):
number(5,j)*(crate(j)-30.5))<=0;
@sum(cai(j):
number(1,j)*(crate(j)-28.5))>=0;
@sum(cai(j):
number(2,j)*(crate(j)-28.5))>=0;
@sum(cai(j):
number(5,j)*(crate(j)-28.5))>=0;
@for(link(i,j):
che(i,j)=number(i,j)/b(i,j));
hehe=@sum(link(i,j): che (i,j));
@for(link(i,j): @gin(number (i,j)));
@for(cai(j):@bin(flag (j)));
hehe<=20;
```

```
ccnum=@sum(cai (j): cnum(j) );
end
```

例7.7

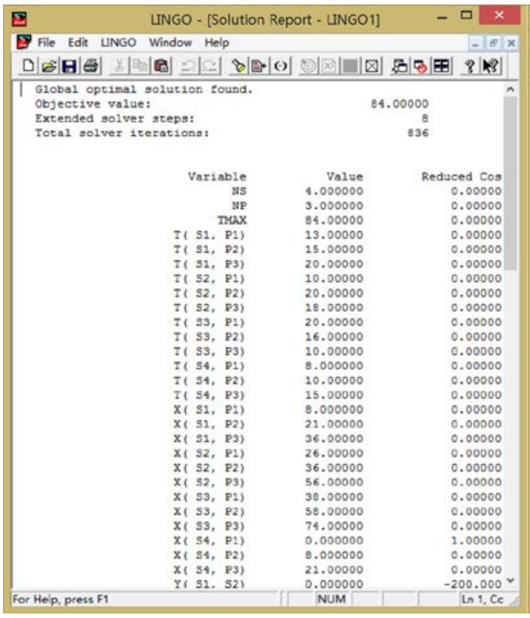
要点：利用LINGO求解——指派问题




```
model:
sets:
    workers/w1..w7/;
    jobs/j1..j7/;
    links(workers,jobs): cost,volume;
endsets
    min=@sum(links: cost*volume);
    @for(workers(I):
        @sum(jobs(J): volume(I,J))=1;
    );
    @for(jobs(J):
        @sum(workers(I): volume(I,J))=1;
    );
data:
    cost= 6 2 6 7 4 2 5
        4 9 5 3 8 5 8
        5 2 1 9 7 4 3
        7 6 7 3 9 2 7
        2 3 9 5 7 2 6
        5 5 2 2 8 11 4
        9 2 3 12 4 5 10;
enddata
end
```

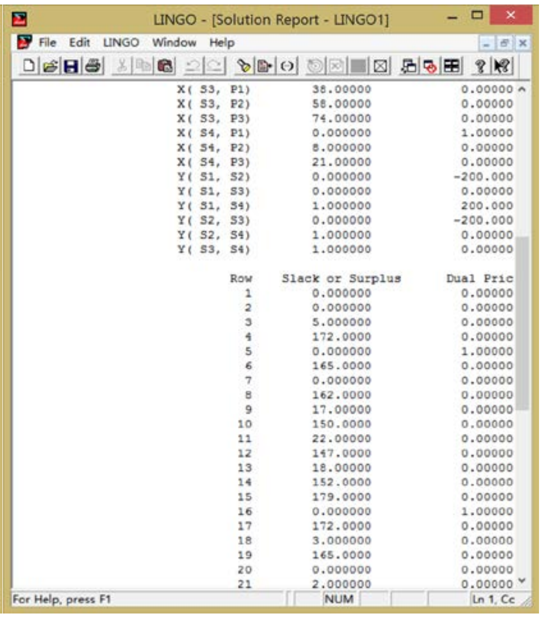
例7.9

要点：利用LINGO求解综合面试问题



```
LINGO - [Solution Report - LINGO1]
  Global optimal solution found.
  Objective value:                              84.00000
  Extended solver steps:                               8
  Total solver iterations:                           836


                       Variable           Value        Reduced Cos
                             NS        4.000000            0.00000
                             NP        3.000000            0.00000
                           TMAX        84.00000            0.00000
                      T( S1, P1)       13.00000            0.00000
                      T( S1, P2)       15.00000            0.00000
                      T( S1, P3)       20.00000            0.00000
                      T( S2, P1)       10.00000            0.00000
                      T( S2, P2)       20.00000            0.00000
                      T( S2, P3)       18.00000            0.00000
                      T( S3, P1)       20.00000            0.00000
                      T( S3, P2)       16.00000            0.00000
                      T( S3, P3)       10.00000            0.00000
                      T( S4, P1)       8.000000            0.00000
                      T( S4, P2)       10.00000            0.00000
                      T( S4, P3)       15.00000            0.00000
                      X( S1, P1)       8.000000            0.00000
                      X( S1, P2)       21.00000            0.00000
                      X( S1, P3)       36.00000            0.00000
                      X( S2, P1)       26.00000            0.00000
                      X( S2, P2)       36.00000            0.00000
                      X( S2, P3)       56.00000            0.00000
                      X( S3, P1)       38.00000            0.00000
                      X( S3, P2)       58.00000            0.00000
                      X( S3, P3)       74.00000            0.00000
                      X( S4, P1)       0.000000            1.00000
                      X( S4, P2)       8.000000            0.00000
                      X( S4, P3)       21.00000            0.00000
                      Y( S1, S2)       0.000000           -200.000
For Help, press F1        NUM        Ln 1, Cc
```



```
LINGO - [Solution Report - LINGO1]
                      X( S3, P1)       38.00000            0.00000
                      X( S3, P2)       58.00000            0.00000
                      X( S3, P3)       74.00000            0.00000
                      X( S4, P1)       0.000000            1.00000
                      X( S4, P2)       8.000000            0.00000
                      X( S4, P3)       21.00000            0.00000
                      Y( S1, S2)       0.000000           -200.000
                      Y( S1, S3)       0.000000            0.00000
                      Y( S1, S4)       1.000000            200.000
                      Y( S2, S3)       0.000000           -200.000
                      Y( S2, S4)       1.000000            0.00000
                      Y( S3, S4)       1.000000            0.00000

                            Row    Slack or Surplus      Dual Pric
                              1        0.000000            0.00000
                              2        0.000000            0.00000
                              3        5.000000            0.00000
                              4        172.0000            0.00000
                              5        0.000000            1.00000
                              6        165.0000            0.00000
                              7        0.000000            0.00000
                              8        162.0000            0.00000
                              9        17.00000            0.00000
                             10        150.0000            0.00000
                             11        22.00000            0.00000
                             12        147.0000            0.00000
                             13        18.00000            0.00000
                             14        152.0000            0.00000
                             15        179.0000            0.00000
                             16        0.000000            1.00000
                             17        172.0000            0.00000
                             18        3.000000            0.00000
                             19        165.0000            0.00000
                             20        0.000000            0.00000
                             21        2.000000            0.00000
For Help, press F1        NUM        Ln 1, Cc
```

```
model:
sets:
    students;
    phases;
    sp(students,phases):t,x;
    ss(students,students) | &1 #LT# &2:y;
endsets
data:
    students = s1..s4;
    phases = p1..p3;
    t=
        13 15 20
        10 20 18
        20 16 10
        8 10 15;
enddata
    ns=@size(students);
    np=@size(phases);
    @for(sp(I,J) | J #LT# np:
        x(I,J)+t(I,J)<=x(I,J+1)
    );
    @for(ss(I,K):
        @for(phases(J):
            x(I,J)+t(I,J)-x(K,J)<=200*y(I,K);
```

```
            x(K,J)+t(K,J)-x(I,J)<=200*(1-y(I,K));
        )
    );
    min=TMAX;
    @for(students(I):
        x(I,3)+t(I,3)<=TMAX
    );
    @for(ss: @bin(y));
End
```

感想：在完成第七部分的练习后，我又找出了大一时在学习数学建模课程期间看过的《数学建模算法与应用》（司守奎、孙玺菁，国防工业出版社）一书，再次阅读此书，与一年前相比我也对于书中的综合问题有了更深刻的认识，对于程序的读写能力也有了很大提高。曾经一些晦涩难懂的模型在经过运筹学学习后变得更容易理解，也体会到了之前如师兄师姐所说的这本书的价值高、内容好。当然，再读此书，书中仍有很多我不熟悉数学公式令人难以理解，不过相信通过之后的学习，一定会对优化类问题有更深刻的认识。