

LINDO使用手册

LINDO Users Manual

第一章 LINDO 初步

LINDO 是 Linear INteractive Discrete Optimizer的缩写，是一个线性规划和整数规划的软件系统。这里介绍的LINDO/386 5.3版本，最大规模的模型的非零系数可以达到1,000,000个，最大变量个数可以达到100,000个，最大目标函数和约束条件个数可以达到32000个，最大整数变量个数可以达到100,000个。它的特点是采用交互方式操作，而且命令简单明了，很容易掌握。用户在输入初步的数学模型后，可以一边运行，一边修改调试，直至获得满意的结果。LINDO既是一个实用的求解大规模线性规划和整数规划系统，也很适用于训练的目的，用来培养学生面对较复杂的实际问题，构造线性及整数规划模型以及求解这些问题的能力。限于本教材的内容，本手册只介绍有关线性规划和整数规划的基本内容，省略了有关二次规划、参数规划以及有关线性规划算法的比较专门的内容。

§ 1.1 LINDO 命令一览表

LINDO 命令纳成十一类，每一类的名称及所属命令如下：

1、查询类(Information)

HELP	求助
COM (Command)	命令列表
LOCAL	当前版本信息
CAT (Categories)	命令分类
TIME	显示当前时间
DATE	显示当前日期

2、文件输入类 (Input)

MAX	键盘输入极大化模型
-----	-----------

MIN	键盘输入极小化模型
RETR (Retrieve)	从磁盘读入模型文件
RMPS (Read MPS)	从磁盘读入MPS文件
TAKE	从磁盘读入模型或命令文件
LEAV (LEAVE)	TAKE命令的终止
RDBC*	读取用SBDC命令保存的基列作为初始基
FBR *	读取用FBS命令保存的基
FINS*	读取用MPS格式保存的基
3、显示类(Display)	
PIC (Picture)	显示矩阵中非零元素的分布及数量级
TABL (Tableau)	显示当前单纯形表
LOOK	显示模型的数学形式
NONZ (Nonzeros)	显示解中的非零变量
SHOC (Show Column)	显示模型中的一列
SOLU (Solution)	显示当前得到的解
RANGE	显示灵敏度分析的结果
BPIC*	显示三角化的当前基逆矩阵
CPRI*	显示所选择列的有关信息
RPRI*	显示所选择行的有关信息
DMPS*	显示以MPS格式表示的解
PPIC*	显示行列重新排列的矩阵，使非零元素最靠近对角线
4、文件输出类 (File Output)	
SAVE	模型存盘
DIVE (Divert)	将屏幕输出转向磁盘文件
RVRT (Revert)	重新回到屏幕输出状态
SMPS (Save MPS)	模型以 MPS 格式存盘
SDBC*	将当前基的列向量以文件形式保存
FBS*	将当前基以文件形式保存
FPUN*	将当前基以MPS形式保存
SMPN*	将当前模型用非标准的MPS格式保存
5、求解类(Solution)	

- | | |
|------------|-----------|
| GO | 求解一个模型 |
| PIV(Pivot) | 进行一次单纯形叠代 |
| GLEXP* | 用字典序方法优化 |
- 6、问题编辑类 (Problem Editing)
- | | |
|-------------------------|---------------|
| ALT(Alter) | 修改模型中的系数 |
| EXT(Extension) | 增加一个约束 |
| DEL(Delete) | 删除一个约束 |
| SUB(Simple Upper Bound) | 给出一个变量的上界 |
| APPC(Append Column) | 增加一列 |
| SLB(Simple Lower Bound) | 给出一个变量的下界 |
| FREE | 取消一个变量的上下界 |
| EDIT | 用全屏幕编辑建立或修改模型 |
- 7、退出 (Quit)
- | | |
|------|-------------|
| QUIT | 退出 LINDO 系统 |
|------|-------------|
- 8、整数规划、二次规划和参数规划 (Integer, Quadratic and Parametric Programs)
- | | |
|-------------------------------|---------------------|
| INT | 定义一个0-1变量 |
| QCP*(Quadratic Programming) | 定义二次规划 |
| PARA*(Parametric Programming) | 求解参数规划 |
| POSD*(Positive Definiteness) | 检查二次规划矩阵的正定性 |
| TITAN* | 紧缩连续变量的上界以及0-1变量的系数 |
| BIP* | 在整数规划中给出一个剪枝的界 |
| GIN (General Integer) | 定义一个整数变量 |
| IPTOL* | 设定整数规划最优解的宽容度 |
- 9、对话参数 (Conversational Parameters)
- | | |
|------------------|----------|
| WIDTH | 设置打印机宽度 |
| TERS (Terse) | 简洁输出信息 |
| VERB (Verbose) | 冗长输出信息 |
| BAT (Batch) | 成批输出信息 |
| PAGE | 设定屏幕页的大小 |
| PAUSE | 暂停 |
- 10、用户子程序

USER	执行用户提供的子程序
11、杂项	
INV* (INVERT)	求当前基的逆矩阵，以减少叠代误差
STAT (STATISTICS)	统计模型矩阵的系数
BUG	如发现系统缺陷，提供有关信息
SET	设置LINDO的内部参数
TITL (TITLE)	给出模型的标题

§ 1.2 LINDO 初步操作

以下命令，可以完成模型输入，模型显示，模型运行，退出LINDO等操作，这些命令是：

- MAX 开始输入一个极大化模型
- MIN 开始输入一个极小化模型
- TITL 输入模型标题
- EDIT 全屏幕输入或全屏幕编辑一个模型
- LOOK 在屏幕上显示已输入的模型
- GO 模型运行
- QUIT 退出LINDO，返回操作系统

在这一章中，先对这几个命令的用法作详细解说。

1、进入LINDO

和调用其他任何应用程序一样，调用LINDO之前，用户先要完成登录、引导操作系统。在系统提示符（在这里，用C>表示系统提示符）出现以后，就可以启动LINDO，即

```
C>LINDO<CR>
```

执行此命令后，屏幕上出现":"。

":"是 LINDO 接受命令的状态，在":"后面，可以键入各种合法的LINDO的命令。每执行一个命令(除QUIT外)，LINDO仍返回这一命令状态，等待下一个命令，如果是QUIT命令，则返回操作系统。

2、输入模型命令MAX / MIN

用MAX / MIN命令可以直接从键盘上输入一个模型到内存。MAX / MIN命令对规模较小的问题比较适用，对大规模问题的建立和输入，将在TAKE命令描述。

设我们要输入的模型的数学形式为：

$$\begin{aligned} & \text{MAX } 5.24X_1+7.3X_2+8.34X_3+4.18X_4 \\ & \text{SUBJECT TO} \\ & \quad 1.5X_1+X_2+2.4X_3+X_4 \leq 2000 \\ & \quad X_1+5X_2+X_3+3.5X_4 \leq 8000 \\ & \quad 1.5X_1+3X_2+3.5X_3+X_4 \leq 5000 \\ & \quad X_1, X_2, X_3, X_4 \geq 0 \end{aligned}$$

输入模型的操作作如下：

```

: MAX  5.24X1+7.3X2+8.34X3+4.18X4  ! "MAX"后面必须空一格
?ST                                     ! "ST"也可以写"SUBJECT TO"
?1.5X1+1.0X2+2.4X3+1.0X4<2000  !用"<"或">"而不能用"<=", ">="
?1.0X1+5.0X2+1.0X3+3.5X4<8000  !?表示当前处于MAX命令状态
?1.5X1+3.0X2+3.5X3+1.0X4<5000  !END结束MAX命令状态。变量非负约束X1, X2,
?END                                     !X3, X4>=0是隐含的，不必也不能输入
    
```

MAX/MIN命令对模型的输入格式有一些限制，这些限制是：

(1)模型中出现的关键词只能是 MAX (或 MIN)，ST (或 SUBJECT TO) 和 END。关键词中不能含有空格。MAX (或 MIN)，ST (或 SUBJECT TO) 的右面至少要有一个空格，关键词中字符大写和小写都合法的。

(2)变量名不超过8个字符，其中第一个字符必须定字母，其余的可以是字母或数字。这样就可以借助变量表示变量的实际含义。例如有10种燃料，可以分别记为 FUEL01，…，FUEL10等。

(3)不等号用"<"表示"≤"，用">"表示"≥"。要注意的是当模型(用 LOOK 命令)输出(到屏幕或打印机)时，不等号分别写成"<="和">="。

(4)目标函数中不能出现常数项，也不能出现等式，例如：

MAX 2X1+3X2+5 和 MAX Z=2X1+3X2都是非法的。

(5)无论连目标函数(第一行)或约束条件中，都不能出现括号、乘号。例如：

MAX 2(X1+X2) 和 MIN 2*X1+3*X2都是非法的。

(6)在约束条件中，变量必须在关系符左面，常数必须在关系符右边，例如：

$3X_1+4X_2-3<0$ 和 $3X_1<3-4X_2$ 都是非法的。

(7)模型中系数必须写成整数或小数形式而不能写成指数形式，例如：下列系数的记法是非法的：.258E+5。

如系数是小数形式，输入的位数限制整数最大6位，小数最大5位。

(8)除关键字中间，系数中间，变量名中间不能嵌入空格外，模型的其他地方都允许出现空格和回车<CR>。在模型输入时，空格和回车是等效的。例如，以下的模型输入格式是有效的：

$$\begin{array}{rcll}
 \max & z = & 3x_1 & +4x_2 & +6x_3 & -5x_4 \\
 \text{s.t.} & & x_1 & +2x_2 & -x_3 & +2x_4 & \leq 10 \\
 & & 3x_1 & +x_2 & -3x_3 & +6x_4 & \leq 25 \\
 & & x_1 & x_2 & x_3 & x_4 & \geq 0
 \end{array}$$

```

:MAX 3X1 + 4 X2+6X3- 5X4<CR>
?ST X1 +2X2-X3<CR>
? +2 X4 <10 3X1+x2-3X3+6X4<25 END
:
    
```

在用MAX/MIN命令输入模型时，如果模型的一行超过屏幕宽度，只要回车就可以在下一行继续输入。

(9)变量的非负约束是隐含的，不需要在模型中加以意义。

3、输入模型标题命令TITL

为了识别模型相应的问题，LINDO设置了TITL命令，可以对模型作简要的说明，TITL放在模型的第一行，内容放在两个引号之间。同时，为了说明约束的性质，也可以在每个约束的前面加上不超过8个字符的说明。例如

```

: TITLE "This is a maximum profit problem"
: MAX 4 X1 + X2 - X3 + 2 X4
? ST
? RESOURCE) X1 - X2 + X3 - X4 < 30
? SALE) 3 X1 + X2 - X3 + 2 X4 < 36
? STORAGE) X1 + 2 X2 + X3 - 2 X4 < 20
    
```

? END

用MAX/MIN命令输入一个模型后，该模型存放在内存中，可以对该模型施加其他命令。内存中只能存放一个模型。如果再次MAX/MIN命令输入另一个模型。内存中的第一个模型将被第二模型代替。如果退出LINDO，内存中有模型也被清除。因此，要保存一个已输入的模型，必须将它存放到磁盘中去，作为磁盘文件保存，模型存盘的命令将在SAVE，SMPS和DIVE命令中讲解。

3.全屏编辑命令EDIT

EDIT命令可以用来输入一个新模型。例如

: EDIT

出现以下编辑窗口，可以在窗口输入、编辑一个模型。输入、编辑完毕以后，按ESC结束编辑，最后一次修改有效；ALT+ESC放弃修改，结束编辑。例如先用EDIT输入以下模型：

```
LINDO/PC 5.3                                © 1995 LINDO SYSTEMS INC.
1      MAX      4 X1 + X2 - X3 + 2 X4
2      ST
3      X1 - X2 + X3 - X4 <=  30
4      3 X1 + X2 - X3 + 2 X4 <=  36
5      X1 + 2 X2 + X3 - 2 X4 <=  20
6      END
...
23
<Esc> - Exit with compile  <Alt><Esc> - Exit without compile
```

用ESC退出编辑并确认修改。

EDIT也可以编辑一个已经输入的模型。例如，如果想把上一个已经输入的模型加上标题，约束加上说明，可以用EDIT进入编辑窗口，将模型编辑修改成：

```
LINDO/PC 5.3                                (C) 1995 LINDO SYSTEMS INC.
1      TITL  "This is a maximum profit problem"
2      MAX      4 X1 + X2 - X3 + 2 X4
3      ST
4      RESOURCE)  X1 - X2 + X3 - X4 < 30
5      SALES)    3 X1 + X2 - X3 + 2 X4 < 36
```

```
6 STORAGE) X1 + 2 X2 + X3 - 2 X4 < 20
7 END
...
23
<Esc> - Exit with compile <Alt><Esc> - Exit without compile
```

用EDIT命令编辑的模型规模由一定限制，最多只能编辑包含65998个字符的模型。规模更大的模型只能用下面介绍的其他命令来编辑，如ALT、EXT、APPC等。

4.模型屏幕显示LOOK

LOOK命令是将内存中的模型的一部分或全部在屏幕上显示。该命令的格式是

: LOOK [行标识]<CR>

其中行标识可以是某一特定的行号，是一个正整数。1表示第1行即目标函数，2表示第一约束等等，行标识也可以是所要显示的符的范围，如2-5表示显示从第二行到第五行止的模型，行标识也可以是"ALL"，即显示整个模型，如果行号n省略，则隐含第一行，即LOOK<CR>和LOOK 1等效。

例如，对上面输入的模型，以下都是合法的LOOK命令：

: LOOK 1<CR>

这时，屏幕显示为：

```
MAX 5.24 X1 + 7.3 X2 + 8.34 X3 + 4.18 X4
```

如 LOOK 命令为：

: LOOK 1-3<CR>

则屏幕显示为：

```
MAX 5.24 X1 + 7.3 X2 + 8.34 X3 + 4.18X4
SUBJECT TO
2) 1.5X1+ X2+2.4X3+ X4<=2000
3) X1+5X2+ X3+3.5X4<=8000
END
```

如 LOOK 命令为：

: LOOK ALL<CR>

则屏幕显示为:

```

MAX  5.24X1 +7.3X2 + 8.34X3 + 4.18X4
SUBJECT  TO
2)1.5X1+ X2 +2.4X3 + X4 <=2000      !不等号的显示与输入形式不同
3) X1+5X2 + X3 +3.5X4 <=8000      !屏幕输出中约束带有行号
4)1.5X1+3X2 +3.5X3 + X4 <=5000
END
:
    
```

5.运行模型命令GO

GO命令的格式为:

```
: GO [n]<CR>
```

其中n为用户指定的最大叠代次数，如果n省略，则隐含值为LINDO指定的最大叠代次数。以上述模型为例，运行操作以及得到的屏幕显示如下:

```

: GO<CR>
  LP OPTIMUM FOUND AT STEP4

  OBJECTIVE FUNCTION VALUE

    1)12737.0600

VARIABLE          VALUE          REDUCED COST
X1                294.117600          .000000
X2                1500.000000          .000000
X3                 0.000000          1.414647
X4                 58.823530          .000000

ROW              SLACK              DUAL PRICES
2)               .000000              1.953529
3)               .000000              .242353
4)               .000000              1.378236

DO SENSITIVE ANALYSIS? ( Y/N ) !如果回答"N", 则出现以下信息
NO. ITERATIONS = 4             !终止运算, 并给出叠代次数
LP OPTIMUM FOUND AT STEP 4
:                               !返回LINDO命令状态
    
```

如果回答"Y", 则给出目标函数系数和右边常数的灵敏度分析结果, 并返回LINDO命令状态。

```
RANGES IN WHICH THE BASIS IS UNCHANGED
```

COST COEFFICIENT RANGES			
VARIABLE	CURRENT COEF	ALLOWABLE INCREASE	ALLOWABLE DECREASE
X1	5.240000	1.029999	.730973
X2	7.300000	3.907058	2.572086
X3	8.340000	1.414647	INFINITY
X4	4.180000	1.343520	.686666
RIGHT HAND SIDE RANGES			
ROW	CURRENT RHS	ALLOWABLE INCREASE	ALLOWABLE DECREASE
2	2000.0000	3000.00000	111.111100
3	8000.0000	1250.00000	166.666700
4	5000.0000	76.92308	1666.667000
:			

5.退出LINDO

命令是QUIT，该命令的功能是退出LINDO，返回操作系统。执行QUIT命令后，未存盘的模型将从内存中清除。用法：

```
:QUIT<CR>
C>
```

下面是一段输入和输出记录，包括以上四个命令以及相应的屏幕输出。"!"右面的是本手册的注释，其他是LINDO的屏幕输出。

```
: MAX 3X1+2X2-X3
? SUBJECT TO
? X1+X2+X3<12
? 2X1+4X2-X3<15
? 3X1-X2+4X3<18
? END
: LOOK ALL
MAX 3 X1 + 2 X2 - X3
SUBJECT TO
  2) X1 + X2 + X3 <= 12
  3) 2X1 + 4X2 - X3 <= 15
  4) 3X1 - X2 + 4X3 <= 18
END
: GO
LP OPTIMUM FOUND AT STEP 2

OBJECTIVE FUNCTION VALUE

1) 19.9285700
```

VARIABLE	VALUE	REDUCED COST	
X1	6.214286	.000000	
X2	.642857	.000000	
X3	.000000	2.642857	
ROW	SLACK	DUAL PRICES	
2)	5.142857	.000000	
3)	.000000	.642857	
4)	.000000	.571429	
NO. ITERATIONS = 2			
GO>DO SENSITIVE ANALYSIS?(Y/N) Y			
RANGES IN WHICH THE BASIS IS UNCHANGED			
COST COEFFICIENT RANGES			
VARIABLE	CURRENT COEF	ALLOWABLE INCREASE	ALLOWABLE DECREASE
X1	3.000000	INFINITY	2.000000
X2	2.000000	3.363636	3.000000
X3	-1.000000	2.642857	INFINITY
RIGHT HAND SIDE RANGES			
ROW	CURRENT RHS	ALLOWABLE INCREASE	ALLOWABLE DECREASE
2	12.000000	INFINITY	5.142857
3	15.000000	18.000000	3.000000
4	18.000000	4.500000	21.750000

第二章 模型文件的存取

以上介绍的四个命令—MAX/MIN, LOOK, GO, QUIT已经可以实现求解一个模型的最基本的操作，但很容易发现，以上操作虽然建立了模型并求得了模型的解，但模型和解都没有生成磁盘文件，因而都无法保存。本节的目的是了解与LINDO模型及求解结果存盘及读取有关的命令，这些命令是

模型文件存盘命令 SAVE

模型文件读取命令 RETR、TAKE

输出转向读取命令 DIVE

恢复屏幕显示命令 RVRT

§ 2.1 模型存盘命令 SAVE

1.命令功能：把已输入内存的模型以磁盘文件的形式存盘

2.命令格式：

 : SAVE [文件名.扩展名]<CR>

其中文件名由至多六个字符组成，第一个必须是字母。扩展名由至多三个符组成。

显然，SAVE命令只有当内存中已输入一个模型时才有效，并且执行SAVE命令后，对内存中的模型没有任何影响。

3.命令使用：设当前内存中已输入一个模型，如键入以下命令：

 : SAVE MODEL.LIN<CR>

则该模型将以MODEL.LIN为名的文件存在磁盘中。如果用户键入的文件名为MODEL，则生成的磁盘文件名为MODEL，扩展名缺省。

这时模型运行的结果还是仅仅在屏幕上显示而没有生成的磁盘文件。关于如何把运行结果生成磁盘文件，将在命令DIVE中讲解。

至于模型磁盘文件是否确实已经生成，可以用操作系统文件目录列表命令DIR来确认。但如果试图用操作系统文件显示命令TYPE来显示该文件的内容，屏幕上将

显示一些无法读认的码，这是因为SAVE命令生成的文件是二进制码的文件，因而无法显示读认，更无法对它进行编辑。

§ 2.2 模型文件读取命令 **RETR (RETRIEVE)**

1.命令功能：从磁盘中读取一个用SAVE命令存盘的模型文件到内存中

2.命令格式：

：RETR[文件名·扩展名]<CR>

其中文件名(扩展名)必须是用SAVE命令生成的磁盘文件名(扩展名)。

3.命令使用：假定当前磁盘中已生成一个名为MODEL.LIN的文件，则可以进行以下操作：

：RETR MODEL.LIN<CR>

该命令执行后，相应的模型将被读到内存中，如果在此之前内存中已有一个模型的话，内存中原有的模型将被清除。对新的模型可以执行显示，运行等操作。

上面提到的SAVE命令只能将模型文件存盘，而不能将运行后的结果存盘。如果要把运行命令GO执行后，显示在屏幕上的运行结果生成磁盘文件，就必须转向命令DIVE。

§ 2.3 输出转向命令 **DIVE (DIVERT)**

1.命令功能：将屏幕输出的信息转到指定的磁盘文件中。

2.命令格式：

：DIVE[文件名·扩展名]<CR>

其中文件名(扩展名)是用户给定的用来存放屏幕输出信息的磁盘文件名。

3.命令使用：

DIVE命令执行以后，一切屏幕输出信息都将转到指定的磁盘文件中，大多数输出信息都将不再在屏幕上显示，但一些重要的揭示信息仍将在屏幕上显示，请看下面的操作：

```

: MAX 2X1+3X2
MAX>ST
MAX>X1+2X2<12
MAX>2X1+4X2<18
END
: DIVE SOLU.DAT
: LOOK ALL
: GO
    LP OPTIMUM FOUND AT STEP  2

    OBJECTIVE FUNCTION VALUE

        1)18.0000000

DO RANGE(SENSITIVITY) ANALYSIS ?
GO>N
: QUIT

C>

```

如果继续操作，由于这些操作在DIVE命令执行之后实施，原来应该在屏幕上显示的信息都将转到磁盘文件SOLU.DAT中去。

现在，如果我们在操作系统下用操作系统文件显示命令TYPE来显示磁盘文件SOLU.DAT的内容，将会看到：

```

C>TYPE SOLU.DAT

MAX  2  X1  +  3  X2
SUBJECT TO
  2)   X1  +  2  X2  <=  12
  3)   2  X1  +  4  X2  <=  18
END

    LP OPTIMUM FOUND AT STEP  2

    OBJECTIVE FUNCTION VALUE

        1)18.0000000

VARIABLE          VALUE          REDUCED   COST
X1                 9.000000          .000000
X2                 .000000          1.000000

ROW                SLACK          DUAL     PRICES

```

```
2)          3.000000      .000000
3)          .000000      1.000000

NO. ITERATIONS=  2

:
```

由于命令DIVE生成的磁盘文件是以ASC II码存放的，因此所生成的文件是可以显示的，可以用各种文本编辑程序(如DOS编辑命令EDIT，Windows中的Notepad等)编辑修改。

这样，我们已经可以对一个模型进行输入(包括键盘输入和读取磁盘文件)、显示、运行、模型存盘、运行结果存盘等基本操作了。

§ 2.4 恢复屏幕显示命令 RVRT (Revert)

1.命令功能：清除输出转向命令DIVE，恢复屏幕输出。

2.命令格式：

```
: RVRT<CR>
```

3.命令使用：从上一节可以知道，命令DIVE执行以后，随后的各种命令，如GO，LOOK等产生的屏幕输出信息都将不再在屏幕上显示而是转到用户指定的磁盘文件中。如果执行一个RVRT，就可以终止DIVE命令的转向功能，恢复信息的屏幕输出。因RVRT命令的这一功能可以知道，RVRT命令一定要与DIVE配合使用，而且RVRT命令一定要在DIVE后面执行。恰当地使用DIVE和RVRT命令，就可以根据用户的要求，把所需要的输出结果存放到磁盘文件中。

§ 2.5 读取命令文件命令 TAKE

1.命令功能：从磁盘读取一个由LINDO命令组成的文件到内存并逐条执行这些命令。

2.命令格式：

```
: TAKE[文件名.扩展名]<CR>
```

其中文件名和扩展名都由用户定义。

3.命令使用：

本手册前面所讲的所有命令，都是由用户从键盘输入的，TAKE命令实际上是把一个命令文件定义为虚拟终端(键盘)，LINDO从该文件读取并执行命令。由此可知，TAKE命令使LINDO命令成批执行。

所谓"命令文件"是一个由若干条合法的LINDO命令组成的文件，它应当用编辑程序（如DOS中的EDIT，WINDOWS中的NOTEPAD等）事先编辑好并存放在磁盘中。命令文件的最后一个命令必须是LEAVE。

例如，先建立如下一个文件名为STREAM.BAT的命令文件：

```
RETR MODEL.LIN
DIVE SOLU.DAT
LOOK ALL
PAUS
GO
N
LEAVE
```

先进入LINDO，然后执行以上批命令STREAM .BAT，则LINDO将依次执行其中的每一条命令，执行到暂停命令PAUS，则暂停执行。键入回车，继续执行。直至遇到批命令终止命令LEAVE，退出批命令，返回LINDO命令状态。

TAKE命令除了可以实现命令的批处理以外，还可以用来把一个由ASCII码写成的模型文件输入LINDO。例如，先用编辑程序写成以下磁盘文件，文件名为MODEL.DAT：

```
MAX 3X1+4X2+X3-X4
ST X1-2X2+X3+3X4<8           !注意，命令文件中模型格式必须
2X1+X2-2X3-2X4<13          !与键盘输入时完全一样，即不带
X1+3X3+X4<21                !行号，不等号为"<"或">"。
END
LEAVE                          !最后一个命令必须是LEAVE
```

先进入LINDO，然后用TAKE命令读取这个文件，就可以将这个模型装入内存。

```
: TAKE MODEL.DAT<CR>           !将文件中的模型读入内存
```

用TAKE命令读取模型文件的方法对大规模模型特别适用。大规模的模型文件可以用其他编辑系统（如DOS的EDIT，Windows的Notepad等）生成，经过编辑，校正，可以减少错误，保证输入正确。有些大型的模型，也可以用用户编制的程序生

成符合LINDO模型格式的模型文件，然后由TAKE命令读入，用LINDO求解。因此，TAKE命令也可以作为LINDO和其他应用程序之间的接口。

当然，我们也可以把模型以及准备对该模型实施的LINDO命令组成一个命令文件。这样当TAKE命令读取这一命令文件后，不仅将模型读入内存，而且将逐打执行文件中的命令，实现模型输入，执行批命令一次完成。例如，命令文件：

```
MAX 3X1+4X2+X3-X4
ST X1-2X2+X3+3X4<8
2X1+X2-2X3-2X4<13
X1+3X3+X4<21
END
LOOK ALL
GO
N
LEAVE
```

用TAKE命令读入后，首先将模型装入内存(MAX命令)，然后在屏幕上显示这个模型(LOOK ALL命令)，运行这个模型(GO命令，在执行中对屏幕提示的回答是N)，最后退出命令文件(LEAVE命令)，返回LINDO命令状态。

§ 2.6 退出命令文件命令 LEAVE

这个命令的功能及格式已在TAKE中讲了。

第三章 线性规划模型的求解

学习了第一章的内容之后，我们已经能够初步利用LINDO来求解一个线性规划模型，得到模型的最优解，亦即在正确输入模型的基础上，用GO命令运行模型。但是仅仅掌握这些基本的操作技术还无法满足多种多样的实际需要。例如有时需要了解进行一次迭代运算的基变化情况，有时不但要求得最优解，而且还要进行灵敏度分析等等。诸如此类线性规划模型求解中在更深层次的问题，便是本章讨论的内容。

§ 3.1 GO 命令的进一步介绍

GO命令除了第一章中介绍的求解模型的最常见功能之外，还有如下几种形式的用途：

(1).对当前内存中的模型进行有限迭代。在用GO命令求解模型时，如果省略迭代次数n，一般情况下计算机只会显示出最后一次运行的结果，这也就是第一章中已提到过的情形。如果欲要了解当前内存中模型进行指定迭代次数时的中间结果，可以使用命令中参数n来达到。例如对于下述问题：

$$\begin{aligned}
 \max \quad & 2x_1 + 8x_2 + 6x_3 \\
 \text{st.} \quad & 8x_1 + 3x_2 + 2x_3 \leq 250 \\
 & 2x_1 + x_2 \leq 50 \\
 & 4x_1 + 3x_3 \leq 150 \\
 & x_1, x_2, x_3 \geq 0
 \end{aligned}$$

当取GO命令中的n为整数2时，计算机显示的内容为：

```

: MAX 2x1+8x2+6x3
? ST
? 8x1+3x2+2x3<250
? 2x1+x2<50
? 4x1+3x3<150
? END
: GO 2                                !运行，最大叠代次数为2次
PIVOT LIMIT OF 2 EXCEEDED. HOW MANY MORE ALLOWED?
GO>                                    !叠代次数2次已超过，允许再叠代多少次？

```

此时尚未求得最优解，如要了解内存中模型第二次迭代的有关数据，可用TABL等显示命令来实现(见第五章的介绍)。

2.追加迭代次数。对于一些规模较大的模型(包括某些尽管空间占有不大，但迭代时间较长的模型)，在使用缺省n的GO命令以后，LINDO可以在经过若干次数的迭代之后，暂时中断，给出类似上一节的提示信息，例如对某一模型有如下求解过程：

```

: MAX 2x1+8x2+6x3
? st
? 8x1+3x2+2x3<250
? 2x1+x2<50
? 4x1+3x3<150
? END
: GO 2                                !运行，最大叠代次数为2次
PIVOT LIMIT OF 2 EXCEEDED. HOW MANY MORE ALLOWED?
GO>3                                    !此时尚未求出最优解，将追加的迭代次数输入

      LP OPTIMUM FOUND AT STEP 2
      OBJECTIVE FUNCTION VALUE

          1)700.000000

VARIABLE      VALUE      REDUCED COST
X1              .000000      22.000000
X2              50.000000       .000000
X3              50.000000       .000000

ROW           SLACK      DUALP RICES
2)              .000000       .000000
3)              .000000       8.000000
4)              .000000       2.000000
NO. ITERATIONS=2
DO RANGE (SENSITIVITY) ANALYSIS?
GO>N
:
    
```

§ 3.2 迭代计算命令 PIV (Pivot)

1.命令功能：对当前基进行一次旋转运算。

2.命令格式：

: PIV [变量名或变量序号]<CR>

3.命令使用：PIV命令执行以后，指定的变量就调入当前基中(即成为进基变量)，离基变量按单纯形算法规则确定。如变量名缺省，则按进基规则依次选择进基变量。例如对于上一节的模型，其操作结果有：

```

: PIV
X2 ENTERS AT VALUE 50.000 IN ROW 3 OBJ. VALUE = 400.00
: PIV
X3 ENTERS AT VALUE 50.000 IN ROW 4 OBJ. VALUE = 700.00
: PIV
    LP OPTIMUM FOUND AT STEP 2

    OBJECTIVE    FUNCTION    VALUE
      1)700.000000

VARIABLE          VALUE          REDUCED COST
X1                 .000000         22.000000
X2                 50.000000          .000000
X3                 50.000000          .000000

ROW                SLACK            DUAL PRICES
2)                 .000000          .000000
3)                 .000000          8.000000
4)                 .000000          2.000000
NO. ITERATIONS =   2
DO RANGE(SENSITIVITY) ANALYSIS ?
PIV>
:

```

此时得到了问题的最优解。如果继续计算，让X1进基，有：

```

: PIV X1
X1 ENTERS AT VALUE 25.000 IN ROW 3 OBJ. VALUE = 150.00
:

```

由于松弛变量是LINDO自动添加的，没有变量名，例如在上一节的模型

$$\begin{array}{lllll}
 \max & 2x_1 & +8x_2 & +6x_3 & \\
 \text{s.t.} & 8x_1 & +3x_2 & +2x_3 & \leq 250 \\
 & 2x_1 & +x_2 & & \leq 50 \\
 & 4x_1 & & +3x_3 & \leq 150 \\
 & x_1 & x_2 & x_3 & \geq 0
 \end{array}$$

中，如果要选第一个松弛变量进基，则需要用这个变量的序号，即第4个变量进基：

```

MAX      2 X1 + 8 X2 + 6 X3
SUBJECT TO
      2)  8 X1 + 3 X2 + 2 X3 <= 250
      3)  2 X1 + X2 <= 50
      4)  4 X1 + 3 X3 <= 150

END

```

```

: PIV 4
SLK 2 ENTERS AT VALUE 250.00 IN ROW 2 OBJ. VALUE= 0.00000E+00
:
    
```

使用PIV命令可以实现如下目的：

(1).一步步地完成单纯形算法的计算。如果将PIV命令与TABL命令交替使用，就可以观察到单纯形表的变化过程。

(2).使某一指定变量进基。这一功能在寻找问题的多重最优解时非常有用。

§ 3.3 灵敏度分析

对一个问题的数据资料可能产生的波动作进一步的研究，一般称之为灵敏度分析。线性规划的灵敏度分析是在建立数学模型和求得最优解之后，针对数据资料变化而作出的。LINDO提供了这方面的功能。

且看下列操作过程：

```

: LOOK ALL<CR>
MAX 2X+3Y
SURJECT TO
(2) 4X+3Y<=10
(3) 3X+5Y<=12
END
: GO
LP OPTIMUM FOUND AT STEP 2

OBJECTIVE FUNCTION VALUE

1) 7.45454500

VARIABLE      VALUE      REDUCED COST
X              1.272727   .000000
Y              1.636364   .000000

ROW           SLACK      DUAL PRICES
2)            .000000    .090909
3)            .000000    .545455

NO. ITERATIONS = 2

DO RANGE (SENSITIVITY) ANALYSIS ?      !是否要灵敏度分析
GO>Y                                     !是
    
```

RANGES IN WHICH THE BASIS IS UNCHANGED

COST COEFFICIENT RANGES

VARIABLE	CURRENT COEF	ALLOWABLE INCREASE	ALLOWABLE DECREASE
X	2.000000	2.000000	.200000
Y	3.000000	.333333	1.500000

RIGHT HAND SIDE RANGES

ROW	CURRENT RHS	ALLOWABLE INCREASE	ALLOWABLE DECREASE
2	10.000000	6.000000	2.800000
3	12.000000	4.666667	4.500000
:			

第四章 模型编辑

用MAX/MIN命令输入模型时，LINDO不支持全屏幕编辑功能，要对输入的模型进行修改，就必须使用专门的命令来实现。本章介绍与模型编辑有关的五个命令，它们是：参数修改命令ALT(Alter)、行增加命令EXT (Extend)、行删除命令DEL (Delete)、变量上限定义命令SUB(Simple Upper Bound)、列增加命令APPC (Append Column)、变量下限定义命令SLB(Simple Lower Bound)，取消变量上下限命令FREE和模型编辑命令EDIT（已在第一章中介绍）。

§ 4.1 参数修改命令 ALT (Alter)

1.命令功能：修改内存中模型的参数。

2.命令格式：

```
: ALT [n] <CR>
```

其中ALT是ALTER的缩写，n代表欲要修改的方程行号。

3.命令使用：在ALT命令下，屏幕上会出现下列提示：

```
VAR:
ALT>变量名<CR>
```

此时如果以合法的变量名作为应答，屏幕上将会继续出现新的提示。

```
NEW COEFFICIENT
ALT>常数<CR>
```

这里的常数即为对应变量的新系数。且看下列实际操作过程及结果。

```
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18
END
```

!注意：目标函数中X2的系数为3

```

: ALT 1                !修改第一行（目标函数行）
VAR:                  !哪个变量
ALT>X2                !X2
NEW COEFFICIENT:     !新的系数：
ALT>5                 !新的系数为5
: LOOK ALL
MAX 2X1+5X2           !目标函数中x2的系数已改为5
SUBJECT TO
2)  X1+2X2<=12
3)  2X1+4X2<=18
END
:

```

除了上述基本用法之外，ALT语句还有以下功能：

(1).修改约束方程右边常。在LINDO提示输入要修改的变量名时，如果键入三个字母"RHS"及<CR>，意即要改变约束方程右边常数。

(2).改变约束方程式的关系符。在LINDO要求输入欲要修改的变量名时，如果键入三个字母"DIR"及<CR>，接着在屏幕提示下输入新的关系符<.>或=，即可实现相应目的。

(3).修改极大极小类型。如果用ALT语句修改的是第一行，键入DIR及<CR>后，即可在提示下输入MAX或MIN来改变原来的目标函数极大极小类型。

(4).增添新的变量。如果要在某一约束方程中增添一个新的变量，亦可仿照上述基本操作进行。因为是新出现的变量，屏幕上会出现有关确认与否的提示信息。以上四点内容综合举例如下：

修改右边常数的例子如下：

```

: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2)  X1+2X2<=12        !原来右边常数是12
3)  2X1+4X2<=18
END

: ALT 2                !修改第二行
VAR:                  !哪个变量
ALT>RHS              !修改右边常数
NEW COEFFICIENT:     !请输入新的系数
ALT>15               !新的系数是15
: LOOK ALL

```

```
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=15           !新的右边常数是15
3) 2X1+4X2<=18
END
:
```

修改约束不等号的方向：

```
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12           !注意：这个约束的不等号方向是“<”
3) 2X1+4X2<=18
END
: ALT 2                   !修改第二行
VAR:                       !哪个变量
ALT>DIR                   !修改第二行不等号方向
NEW DIRECTION:           !新的方向
ALT> >                   !新的方向是“>”
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2 >= 12
3) 2X1+4X2 <= 18
END
:
```

修改目标函数优化方向：

```
: LOOK ALL
MAX 2X1+3X2               !注意：原来的优化方向是极大化
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18

END
: ALT 1                   !修改第一行（目标函数）
VAR:                       !输入要修改的变量
ALT>DIR                   !修改优化方向
NEW DIRECTION:           !新的方向
ALT>MIN                   !极小化
: LOOK ALL
MIN 2X1+3X2               !现在的优化方向是极小化
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18
```

```
END
```

```
:
```

增加一个新的变量以及相应的系数：

```
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18
END
: ALT 1          !修改第一行
VAR:            !请输入变量
ALT>X3         !变量为X3
VARIABLE NOT USED IN THIS PROBLEM BEFORE. WANT IT INCLUDED?
ALT>Y          !以前未曾用到这个变量，是否将其引入？（是）
NEW COEFFICIENT: !新的系数
ALT>4          !系数为4
: LOOK ALL
MAX 2X1+3X2+4X3
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18
END
:
```

使用ALT命令时须注意下述两个问题：

(1).不可把一行中所有的变量系数都改为零，否则模型结构将被破坏(出现非法约束行)。如果要想删除一行，应使用DEL命令(见本章第三节)。

(2).执行ALT命令后再次迭代运算得到的最优解，是在上一次的最后结果基础上根据新参数修改情况进行的，而不是从头开始计算。因此，在求解大规模的问题时，用ALT命令可以较快地得到多种方案的最优解。

§ 4.2 行增加命令 EXT (Extension)

1.命令功能：在当前模型中增加若干约束(行)。

2.命令格式：

```

: EXT<CR>
? .....
? .....
? END

```

3.命令使用：在EXT命令下对当前模型增加有约束是追加在原有约束的最后一个之后的，其键盘操作方法类同于MAX/MIN命令下的要求。请看下例：

```

: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18
END
: EXT                !增加行
BEGIN EXTEND WITH ROW 4 !从第四行开始增加
? 3X1+5X2<15
? END
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12
3) 2X1+4X2<=18
4) 3X1+5X2<=15
END
:

```

使用行增加命令时须注意，结束该命令状态应使用END命令。

§ 4.3 行删除命令 DEL (Delete)

1.命令功能：从当前模型中删除一个约束(行)。

2.命令格式：

```

: DEL [n]<CR>

```

其中n表示要删除的约束的行号。

3.命令使用：请看下列：

```

: LOOK ALL
MAX 2X1+3X2
SUBJECT TO

```

```

2)  X1+2X2<=12
3)  3X1+5X2<=15
4)  2X1+4X2<=18
END
: DEL 2                !删除第二行
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2)  3X1+5X2<=15
3)  2X1+4X2<=18
END
: DEL 2                !删除第二行。注意，这时的第二行
: LOOK ALL            !是原来问题的第三行
MAX 2X1+3X2
SUBJECT TO
2)  2X1+4X2<=18
END
:

```

关于DEL命令的使用有如下三点说明：

(1).第一行的目标函数不能用DEL命令删除。如果出现删除第一行的命令，屏幕上会出现提示信息，如：

```

: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2)  X1+2X2<=12
3)  3X1+5X2<=15
4)  2X1+4X2<=18
END
: DEL 1                !删除第一行
CANNOT DELETE ROW 1 ...REENTER ROW NUMBER
ROW:                  !不能删除第一行，重新输入行号
:

```

(2).如果输入的行号超过已有的编号，屏幕上会出现提示信息。例如对于本节的模型，有：

```

: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2)  X1+2X2<=12
3)  3X1+5X2<=15
4)  2X1+4X2<=18

```

```

END
: DEL 5                !删除第5行
INVALID ROW NUMBER...REENTER VALID ROWS ARE FROM 1 TO 4
ROW:                  !无效的行号，重新输入有效的1~4行行号
:
    
```

(3).将原模型中的某一行删除后，LINDO会重新按自然数顺序排列剩下的约束，如果忽略了这一点，则有可能删去本应保留的约束。

§ 4.4 变量上限定义命令 SUB (Simple Upper Bound)

1.命令功能：给当前模型中的变量设置上限。

2.命令格式：

```
: SUB [变量名] [常数]<CR>
```

其中的常数为给相应变量设置的上限值。

3.命令使用：给下述模型中变量X1设置上限5（即 $x_1 \leq 5$ ）的操作如下：

```

: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2)  X1+2X2<=12
3)  3X1+5X2<=15
4)  2X1+4X2<=18
END
: SUB X1 5                !为x1设定上限5
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2)  X1+2X2<=12
3)  3X1+5X2<=15
4)  2X1+4X2<=18
END
SUB X1 5.00              !x1的上限为5

: LOOK ALL

MAX      2 X1 + 3 X2
SUBJECT TO
      2)  X1 + 2 X2 <=  12
      3)  3 X1 + 5 X2 <=  15
      4)  2 X1 + 4 X2 <=  18

END
    
```

SUB	X1	5.00000
:		

虽然给一个变量增加上限与增加一个相应的约束作用是一样的，但增加变量上限的计算效率要高得多，因此凡是遇到有上限变量得情况，都应该用SUB命令而避免增加约束。

§ 4.5 列增加命令 APPC(Append Column)

1.命令功能：在当前模型中增加一个变量并增加相应的列。

2.命令格式：

: APPC [变量名]<CR>

其中的变量名是要在模型中新增加的变量。

3.命令使用：在APPC命令状态下，屏幕上会出现一系列提示信息，只要按照提示要求依次输入行号和对新的变量系数即可。请看下例：

```

: LOOK ALL
MAX 2X1+3X2

SUBJECT TO
2) X1+2X2<=12
3) 3X1+5X2<=15
4) 2X1+4X2<=18
END
: APPC X3           !增加列X3
APPC>1 7           !X3在第一行中的系数是7
APPC>2 8           !X3在第二行中的系数是8
APPC>3 9           !X3在第三行中的系数是9
APPC>0             !退出APPC命令
: LOOK ALL
MAX 2X1+3X2+7X3
SUBJECT TO
2) X1+2X2+8X3<=12
3) 3X1+5X2+9X3<=15
4) 2X1+4X2<=18
END
:

```

关于APPC命令的使用，有下列三个须注意的问题：

(1).如果在APPC命令格式中缺省变量名，则屏幕上会出现有关要求输入变量名的提示信息。这时再输入新变量名。

(2).如果输入了错误的行号，屏幕上会出现警告信息。例如对于本节的实例，最大行号为了当错误地输入行号4时，就会显示如下内容：

```
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12
3) 3X1+5X2<=15
4) 2X1+4X2<=18
END
: APPC X3
APPC>1 7
APPC>2 8
APPC>3 9
APPC>5 10
INVALID ROW DISREGARDED
APPC>
:
```

§ 4.6 变量下限定义命令 SLB (Simple Lower Bound)

1.命令功能：给当前模型中的变量设置下限。

2.命令格式：

 : SLB [变量名] [常数]<CR>

其中的常数为给相应变量设置的下限值。

3.命令使用：给下述模型中变量X1设置下限1.5（即 $x_1 \geq 1.5$ ）的操作如下：

```
: LOOK ALL
MAX 2X1+3X2
SUBJECT TO
2) X1+2X2<=12
3) 3X1+5X2<=15
4) 2X1+4X2<=18
END
: SLB X1 1.5           !为x1设定下限1.5
: LOOK ALL
MAX 2X1+3X2
```

```

SUBJECT TO
2)  X1+2X2<=12
3)  3X1+5X2<=15
4)  2X1+4X2<=18
END
SLB X1 1.5000          !x1的下限为1.5
:
    
```

请注意，变量下限可以是正数，也可以是负数，当然也可以是零。因为 LINDO 默认变量是非负的，即所有变量的下限都是 0，因此没有必要把变量的下限再设为 0。

§ 4.7 取消变量上下限命令 FREE

1.命令功能：取消当前模型中的变量的上限和下限。

2.命令格式：

```
: FREE [变量名]<CR>
```

3.命令使用：取消上一节模型中变量X1的上下限：

```

: LOOK ALL
MAX    2 X1 + 3 X2
SUBJECT TO
    2)  X1 + 2 X2 <=  12
    3)  3 X1 + 5 X2 <=  15
    4)  2 X1 + 4 X2 <=  18
END
SUB      X1          5.00000          ! x1 的上限为 5（隐含的下限为 0）

: FREE X1          ! 取消 x1 的上下限
: LOOK ALL

MAX    2 X1 + 3 X2
SUBJECT TO
    2)  X1 + 2 X2 <=  12
    3)  3 X1 + 5 X2 <=  15
    4)  2 X1 + 4 X2 <=  18
END
FREE      X1          ! 变量 x1 的下限为 -∞，上限为 ∞
:
    
```

请注意，如果只要取消变量的上限而保留下限 0，则需要再加上下限 0：

```

: SLB X1 0          ! x1 的下限设为 0
: LOOK ALL
    
```

```
MAX      2 X1 + 3 X2
SUBJECT TO
    2)   X1 + 2 X2 <=  12
    3)   3 X1 + 5 X2 <=  15
    4)   2 X1 + 4 X2 <=  18
END                                             ! 所有变量下限为 0, 上限为∞
:                                             ! 即恢复原来的设置
```

§ 4.8 模型全屏幕编辑命令 **EDIT**

这个命令已经在第一章中讲过了。

第五章 数据信息的显示

第一章已介绍过如何用LOOK命令来察看内存中的模型，LOOK命令是LINDO中最基本的显示类命令。但是仅仅掌握这一命令还远远不敷需要，诸如了解单纯形表变化情况，查询计算结果等情况，就要用到其它一些显示命令。

本章将要介绍的六种显示类命令是单纯形表显示命令TABL (Tableau)。矩阵非零元素显示命令PIC (Picture)。列显示命令SHOC(Show Column)。答案显示命令SOLU(Solution)。非零解显示命令NONZ(Nonzeros)和灵敏度分析结果显示命令RANGE。

§ 5.1 单纯形表显示命令 TABL (Table)

1.命令功能：显示当前单纯形表。

2.命令格式：

: TABL<CR>

3.命令使用：在线性规划模型的迭代过程中，如果使用了TABL命令，则屏幕上显示出当前有单纯形表。因此，PIV命令和TABL命令交替使用，可以一步一步地完成并观看到单纯形的算法过程，这对于了解基的变化情况很有用。下面将第一章已用GO命令求出结果的模型用PIV命令和TABL命令重新举例如下：

```
: LOOK ALL
```

```
MAX 2X1+3X2
```

```
SUBJECT TO
```

```
2) X1+2X2<=12
```

```
3) 2X1+4X2<=18
```

```
END
```

```
: TABL
```

```
THE TABLEAU
```

ROW(BASIS)		X1	X2			
1	ART	-2.000	-3.000	.000	.000	.000
2	SLACK	1.000	2.000	1.000	.000	12.000
3	SLACK	2.000	4.000	.000	1.000	18.000

: PIV
 X2 ENTERS AT VALUE 4.5000 IN ROW 3 OBJ. VALUE = 13.500

: TABL

THE TABLEAU

ROW(BASIS)		X1	X2			
1	ART	-.500	.000	.000	.750	13.500
2	SLACK	.000	.0001	.000	-.500	3.000
3	X2	.500	1.000	.000	.250	4.500

: PIV

X1 ENTERS AT VALUE 9.0000 IN ROW 3 OBJ. VALUE = 18.000

: TABL

THE TABLEAU

ROW(BASIS)		X1	X2			
1	ART	.000	1.000	.000	1.000	18.000
2	SLACK	.000	.000	1.000	-.500	3.000
3	X1	1.000	2.000	.000	.500	9.000

: PIV

LP OPTIMUM FOUND AT STEP 2

OBJECTIVE FUNCTION VALUE

1)18.0000000

VARIABLE	VALUE	REDUCED COST
X1	9.000000	.000000
X2	.000000	1.000000

ROW	SLACK	DUAL PRICES
2)	3.000000	.000000
3)	.000000	1.000000

NO. ITERATIONS=2

DO RANGE(SENSITIVITY) ANALYSIS ?

PIV>N

: TABL

THE TABLEAU

ROW (BASIS)		X1	X2			
1	ART	.000	1.000	.000	1.000	18.000
2	SLACK	.000	.000	1.000	-.500	3.000
3	X1	1.000	2.000	.000	.500	9.000

:

§ 5.2 矩阵非零元素显示命令 PIC (Picture)

1.命令功能：显示当前单纯形表的非零系数矩阵。

2.命令格式：

: PIC<CR>

3.命令使用：需要注意的是，PIC命令下显示出的非零系数矩阵，对于两位以上的数据是用特定的英文字母表示的。请看下例：

```

: LOOK ALL
MAX 5X1+12X2+130X3
SUBJECT TO
2) 0.1X1+4.2X2+24X3<=1500
3) 0.05X1+0.3X2+14X3<=2400
END

: PIC

          X      X      X
          1      2      3

1:        5      B      C      MAX
2:        U      A      B      <D
3:        U      T'     B      <D
    
```

其中A代表4.2，B代表12、24和14，C代表130。

下表列出了字母间的一一对应关系。

Z .000000	到	.000001
Y .000001	到	.000009
X .000010	到	.000099
W .000100	到	.000999
V .001000	到	.009999
U .010000	到	.099999
T .100000	到	.999999
A 1.000001	到	10.000000
B 10.000001	到	100.000000
C 100.000001	到	1000.000000
D 1000.000001	到	10000.000000

```
E 10000.000001      到 100000.000000
F 100000.000001     到 1000000.000000
G >1000000
```

§ 5.3 列显示命令 SHOC (Show Column)

1.命令功能：显示当前单纯形表中某一变量的非零系数列矩阵。

2.命令格式：

```
: SHOC[变量名]<CR>
```

3.命令使用：请看下面的例子：

```
: LOOK ALL
MAX 5X1+12X2+130X3
SUBJECT TO
2) 0.1X1+4.2X2+24X3<=1500
3) 0.05X1+0.3X2+14X3<=2400
END

: SHOC X1
ROW      COEF.      DUAL-PRICE
1         5.00000    .000000
2         .100000    .000000
3         .500000E-01 .000000

: SHOC X2
ROW      COEF.      DUAL-PRICE
1        12.0000    .000000
2         4.20000    .000000
3         .300000    .000000

: QUIT
```

§ 5.4 答案显示命令 SOLU (Solution)

1.命令功能：显示求解结果。

2.命令格式：

```
: SOLU<CR>
```

3.命令使用：用SOLU命令显示求解结果，可以节省模型再次运行的时间，使问题分析更加方便。

且看下例：

```

: LOOK ALL
MAX 2X1+8X2+6X3
SUBJECT TO
2) 8X1+3X2+2X3<=250
3) 2X1+X2<=50
4) 4X1+3X3<=150
END

: GO
      LP OPTIMUM FOUND AT STEP  2
      OBJECTIVE FUNCTION VALUE
        1)700.000000
VARIABLE      VALUE      REDUCED COST
X1              .000000      22.000000
X2             50.000000       .000000
X3             50.000000       .000000

ROW           SLACK           DUALP RICES
2)           .000000           .000000
3)           .000000           8.000000
4)           .000000           2.000000

NO. ITERATIONS =  2

DO RANGE(SENSITIVITY) ANALYSIS ?
GO>N

: SOLU
      OBJECTIVE FUNCTION VALUE
        1)700.000000
VARIABLE      VALUE      REDUCED COST
X1              .000000      22.000000
X2             50.000000       .000000
X3             50.000000       .000000

ROW           SLACK           DUAL PRICES
2)           .000000           .000000
3)           .000000           8.000000
4)           .000000           2.000000

NO. ITERATIONS =  2
:

```

§ 5.5 非零解显示命令 NONZ (Non-zero)

1.命令功能：显示当前模型求解结果中的非零解。

2.命令格式：

: NONZ<CR>

3.命令使用：这一命令对于变量很多且零解也很多的问题极为有用。对于上节 SOLU命令中已求出

的解，用NONZ命令显示的内容如下：

```

: NONZ
      OBJECTIVE FUNCTION VALUE

          1) 700.000000
VARIABLE          VALUE          REDUCED COST
X2                 50.000000          .000000
X3                 50.000000          .000000

ROW              SLACK              DUAL PRICES
3)                .000000              8.000000
4)                .000000              2.000000

NO. ITERATIONS=2
:
    
```

如果解答中无零解，则NONZ命令与SOLU命令的功能相同。

§ 5.6 灵敏度分析结果显示命令 RANGE

1.命令功能：显示当前模型的灵敏度分析结果。

2.命令格式：

: RANGE<CR>

3.命令使用：对于第三章第三节中已求得灵敏度分析结果的例子，使用RANGE命令后可显示如下内容：

```

: RANGE

RANGES IN WHICH THE BASIS IS UNCHANGED
    
```

COST COEFFICIENT RANGES			
VARIABLE	CURRENT COEF	ALLOWABLE INCREASE	ALLOWABLE DECREASE
X1	2.000000	22.000000	INFINITY
X2	8.000000	INFINITY	8.000000
X3	6.000000	INFINITY	6.000000
RIGHT HAND SIDE RANGES			
ROW	CURRENT RHS	ALLOWABLE INCREASE	ALLOWABLE DECREASE
2	250.000000	INFINITY	.000000
3	50.000000	.000000	50.000000
4	150.000000	.000000	150.000000
:			

同其它显示类命令一样，RANGE命令不进行计算，只用于显示。因此这一命令只能用在GO命令的灵敏度分析之后。如果不对模型进行计算便使用该命令，则无法显示出需要的信息。且看下例：

```

: RETR MODEL.LIN

: RANGE

WARNING, SOLUTION MAY BE NONOPTIMAL/NONFEASIBLE

RANGES IN WHICH THE BASIS IS UNCHANGED

COST COEFFICIENT RANGES

VARIABLE      CURRENT      ALLOWABLE      ALLOWABLE
              COEF        INCREASE      DECREASE

X1            2.000000    -2.000000    INFINITY
X2            8.000000    -8.000000    INFINITY
X3            6.000000    -6.000000    INFINITY

RIGHT HAND SIDE RANGES

ROW           CURRENT      ALLOWABLE      ALLOWABLE
              RHS        INCREASE      DECREASE

2            250.000000  INFINITY      .000000
3             50.000000  INFINITY      .000000
4            150.000000  INFINITY      .000000
    
```

第六章 整数规划

整数规划是数学规划理论中的重要内容之一。LINDO具有了0-1整数规划和整数规划两种模型求解的功能。LINDO可以将模型中的一部分变量定义为0-1变量、一部分变量定义为整数变量、其余变量仍保留为连续变量。因此，LINDO具有求解0-1、整数混合规划的功能。在使用LINDO求解整数规划的功能时，除0-1变量、整数变量的定义命令之外，其他操作如求解、模型编辑等与前几章中所介绍过的内容大致相同。因此，本章把定义0-1变量和定义整数变量的命令INT、GIN作为主要内容来介绍。

§ 6.1 定义 0-1 变量命令 INT

- 1.命令功能：定义当前模型中的0-1型整数变量。
- 2.命令格式：

: INT[变量名]<CR>

其中的变量名表示将要定义为0-1变量的变量。

3.命令使用：输入一个0-1型整数规划模型的方法，是先按前述的线性规划模型的输入方法进行操作，尔后用INT命令来定义0-1变量。INT命令一般应在对应的线性规划模型输入完毕后立即进行。下面请看有三个0-1变量的整数规划模型的上机操作实例：

```

MAX 3X1+4X2+5X3+3X4+2X5
? ST
? 2X1+3X2+4X3+5X4+X5<11
? 3X1+4X2+2X3+X4+3X5<16
? 7X1+2X2+5X3+4X4+2X5<19
? END
:INT X1           ! 定义x1为0-1变量
:INT X2           ! 定义x2为0-1变量
:INT X3           ! 定义x3为0-1变量
: LOOK ALL
MAX 3X1+4X2+5X3+3X4+2X5
SUBJECT TO
2) 2X1+3X2+4X3+5X4+X5<=11
3) 3X1+4X2+2X3+X4+3X5<=16

```

4) $7X_1+2X_2+5X_3+4X_4+2X_5 \leq 19$

END

INTEGER-VARIABLES=3

! 前三个变量为0-1变量

: GO

LP OPTIMUM FOUND AT STEP 6 ! 已获得线性规划的最优解

OBJECTIVE FUNCTION VALUE

1) 16.4000000

! 线性规划的目标函数值

VARIABLE	VALUE	REDUCED COST
X1	1.000000	.200000
X2	.800000	.000000
X3	1.000000	1.000000
X4	.000000	1.400000
X5	2.600000	.000000

ROW	SLACK	DUAL PRICES
2)	.000000	.800000
3)	.000000	.400000
4)	.200000	.000000

NO. ITERATIONS=6

! 开始分支定界算法

BRANCHES=0 DETERM.=-5.000E0

SET 2 TO 1 AT 1 BND=16.333330 TWIN=16.000000

SET 1 TO 1 AT 2 BND=16.300000 TWIN=16.000000

SET 3 TO 1 AT 3 BND=16.000000 TWIN=16.000000

NEW INTEGER SOLUTION AT BRANCH 3 PIVOT 10

OBJECTIVE FUNCTION VALUE

1) 16.0000000

! 第一个(0-1)整数解

VARIABLE	VALUE	REDUCED COST
X1	1.000000	-1.000000
X2	1.000000	-2.000000
X3	1.000000	-3.000000
X4	.000000	7.000000
X5	2.000000	.000000

ROW	SLACK	DUAL PRICES
2)	.000000	2.000000
3)	1.000000	.000000
4)	1.000000	.000000

NO. ITERATIONS=10

BRANCHES=3 DETERM.=1.000E0

```

BEST REMAINING SOLUTION NO BETTER THAN 16.00000
DELETE 3 AT LEVEL 3
DELETE 1 AT LEVEL 2
DELETE 2 AT LEVEL 1
ENUMERATION COMPLETE. BRANCHES = 3 PIVOTS = 10

LAST INTEGER SOLUTION IS THE BEST FOUND! 以上整数解是最优解
:

```

从上例可以看到，求解一个0-1整数规划模型的命令，仍是GO命令。GO命令下的第一步计算结果不是整数的，接着键入回车键，即开始用分支定界法计算整数解(而不是进行灵敏度分析)。

当模型中有很多变量需要定义成0-1变量，可以有以下命令：

```
: INT <n>
```

其中n表示模型中的0-1型整数变量数目。使用这命令来定义0-1变量，可以极大地提高键盘输入效率。对于上节的例子，三个0-1变量的定义可以如此操作：

```
: INT 3
```

使用该命令时须特别注意变量在目标函数中的排列次序。当目标函数中不是包含全部变量时，应慎用这一命令。命令中的n，是指目标函数中(或模型中先输入的)前n个变量，不注意这一点，就容易出错。

如果要取消已定义的0-1变量，用以下命令：

```
: INT 0<CR>
```

键入这一命令后，模型中原有的0-1变量全部被取消，整数规划模型变成线性规划模型，但原有的 0-1变量被限制在区间[0, 1]上。如：

```

: LOOK ALL
MAX    3 X1 + 4 X2 + 5 X3 + 3 X4 + 2 X5
SUBJECT TO
    2)  2 X1 + 3 X2 + 4 X3 + X4 + 3 X5 <=  11
    3)  3 X1 + 4 X2 + 2 X3 + X4 + 3 X5 <=  16
    4)  7 X1 + 2 X2 + 5 X3 + 4 X4 + 2 X5 <=  19
END
INTE    3
: INT 0
: LOOK ALL
MAX    3 X1 + 4 X2 + 5 X3 + 3 X4 + 2 X5

```

```

SUBJECT TO
    2)  2 X1 + 3 X2 + 4 X3 + X4 + 3 X5 <=  11
    3)  3 X1 + 4 X2 + 2 X3 + X4 + 3 X5 <=  16
    4)  7 X1 + 2 X2 + 5 X3 + 4 X4 + 2 X5 <=  19

END
SUB      X1      1.00000
SUB      X2      1.00000
SUB      X3      1.00000

:

```

§ 6.2 定义整数变量命令 GIN

1.命令功能：定义当前模型中的0-1型整数变量。

2.命令格式：

```
: INT [变量名]<CR>
```

其中的变量名表示将要定义为0-1变量的变量。

定义整数变量以及求解整数（及混合）规划的方法基本上和 0-1 变量相同，仅举以下一个例子说明：

```

:MAX      3 X1 + 4 X2 + 5 X3 + 3 X4 + 2 X5
?ST
?2X1+3X2+4X3+5X4+X5<11
?3X1+4X2+2X3+X4+3X5<16
?7X1+2X2+5X3+4X4+2X5<19
?END
:GIN X1
:GIN X3
:GIN X5
:LOOK ALL

MAX      3 X1 + 5 X3 + 2 X5 + 4 X2 + 3 X4
SUBJECT TO
    2)  2 X1 + 4 X3 + X5 + 3 X2 + 5 X4 <=  11
    3)  3 X1 + 2 X3 + 3 X5 + 4 X2 + X4 <=  16
    4)  7 X1 + 5 X3 + 2 X5 + 2 X2 + 4 X4 <=  19

END
GIN      3

```

: GO

LP OPTIMUM FOUND AT STEP 2
 OBJECTIVE VALUE = 16.8999996
 SET X3 TO >= 2 AT 1, BND= 16.00 TWIN= 16.41 8

NEW INTEGER SOLUTION OF 16.0000 AT BRANCH 1 PIVOT 8
 BOUND ON OPTIMUM: 16.40741

FLIP X3 TO <= 1 AT 1 WITH BND= 16.407408
 SET X3 TO >= 1 AT 2, BND=16.41 TWIN=-0.100E+31 8
 SET X5 TO >= 3 AT 3, BND=16.23 TWIN=16.06 10
 SET X5 TO <= 3 AT 4, BND=16.23 TWIN=-0.1000E+31 10
 SET X1 TO <= 0 AT 5, BND=16.12 TWIN=16.00 12

NEW INTEGER SOLUTION OF 16.1176472 AT BRANCH 3 PIVOT 12
 BOUND ON OPTIMUM: 16.11765

DELETE X1 AT LEVEL 5
 DELETE X5 AT LEVEL 4
 DELETE X5 AT LEVEL 3
 DELETE X3 AT LEVEL 2
 DELETE X3 AT LEVEL 1

ENUMERATION COMPLETE. BRANCHES= 3 PIVOTS= 12

LAST INTEGER SOLUTION IS THE BEST FOUND
 RE-INSTALLING BEST SOLUTION...

OBJECTIVE FUNCTION VALUE

1) 16.11765

VARIABLE	VALUE	REDUCED COST
X1	0.000000	-0.117647
X3	1.000000	-1.823529
X5	3.000000	0.411765
X2	1.235294	0.000000
X4	0.058824	0.000000

ROW	SLACK OR SURPLUS	DUAL PRICES
2)	0.000000	0.470588
3)	0.000000	0.647059
4)	5.294117	0.000000

NO. ITERATIONS= 12
BRANCHES= 3 DETERM.= 1.000E 0

与 0-1 变量问题一样，可以用

: GIN n

定义模型的前 n 个变量为整数变量；用

: GIN 0

取消对所有整数变量的定义。

第七章 MPS 格式文件

前面几章涉及到的模型格式，无论是输入形式还是显示格式，都接近于一般为的书面数学表达方法，因此极易于被人接受，更方便于教学。但是，这样一种接近习惯表达方式的模型格式，有时也会带来不便。例如对变量排列不整齐的较大模型，如果想要了解某一变量出现在哪一个约束中，并且检查它的系数是多少，便要费一番周折。针对诸如此类的问题，LINDO提供了另外一种格式来表达模型，这就是本章将要介绍的MPS格式文件。

§ 7.1 MPS 格式文件生成命令 SMPS

1.命令功能：将当前模型转换成MPS格式并存盘。

2.命令格式：

: SMPS[文件名·扩展名]<CR>

其中的扩展名可以缺省，如缺省，系统将自动给予MPS作为扩展名。

3.命令使用：使用SMPS命令之后，内存中原有的LIN格式文件仍有保留，另外生成新的MPS格式文件。请看下面的操作及讲解：

: LOOK ALL

MAX 3 X1 + 4 X2 + 5 X3 + 3 X4 + 2 X5

```

SUBJECT TO
    2)  2 X1 + 3 X2 + 4 X3 + 5 X4 +   X5 <=  11
    3)  3 X1 + 4 X2 + 2 X3 +   X4 + 3 X5 <=  16
    4)  7 X1 + 2 X2 + 5 X3 + 4 X4 + 2 X5 <=  19

END

: INT X1
: GIN X2
: SUB X3 5
: SLB X4 1.5
: SMPS MPS.LIS
: QIUT
C> TYPE MPS.LIN
NAME                                     ( MAX)
ROWS
  N 1
  L 2
  L 3
  L 4
COLUMNS
  INTEGER1 'MARKER'                    'INTORG'
  X1      1      3.0000000
  X1      2      2.0000000
  X1      3      3.0000000
  X1      4      7.0000000
  X2      1      4.0000000
  X2      2      3.0000000
  X2      3      4.0000000
  X2      4      2.0000000
  INTEGER2 'MARKER'                    'INTEND'
  X3      1      5.0000000
  X3      2      4.0000000
  X3      3      2.0000000
  X3      4      5.0000000
  X4      1      3.0000000
  X4      2      5.0000000
  X4      3      1.0000000
  X4      4      4.0000000
  X5      1      2.0000000
  X5      2      1.0000000
  X5      3      3.0000000
  X5      4      2.0000000
RHS
  RHS      2      11.0000000
  RHS      3      16.0000000
  RHS      4      19.0000000
BOUNDS

```

```
LO LINDOBND X1      0.000000000
UP LINDOBND X1      1.000000000
LO LINDOBND X3      0.000000000
UP LINDOBND X3      5.000000000
LO LINDOBND X4      1.500000000
ENDATA
```

可见，一个用MPS格式文件由模型名、行信息、变量信息、常数列信息、变量上界信息和文件结束符六段组成。

第一段是模型名，只包括一行。其中NAME是关键字，其余由用户输入。如果MPS文件是由LINDO自动产生的，则自动生成模型名就是输入模型的标题TITL。

行信息段由关键字“ROWS”开始，以下每一行说明模型的一行。每一行由行标识和行号组成。行标识符为：

- N 目标函数
- L “小于等于”约束
- E “等于”约束
- G “大于”约束

行号以ROW开头，后继以行编号，如ROW10001，ROW10002等。

MPS格式文件的一个重要优点是可以定义一个以上的目标函数，即在行信息段中可以出现一个以上的以“N”为标识的行。例如：

```
NAME example of a model with muliti-objective functions
ROWS
N ROW10001
N ROW10002
L ROW10003
E ROW10004
G ROW10005
COLUMNS
.....
```

当一个多于目标函数的MPS格式文件读入后，系统将会提示目标函数行号，要求用户确认其中一个目标函数作为当前运行的目标函数。在运行的结果中，系统将会输出当前最优解时各目标函数的值。这一功能为多目标评价提供了很大方便。

MPS格式文件的变量信息段由关键字“COLUMNS”开始，每一行中包括：变量名（即系数矩阵的列）、行号（即系数矩阵的行）以及该变量在该行中的系数。矩阵中的零元素不进行描述。

第四段是右边常数段，由关键字“RHS”开始，其中每一行包括：“RHS”、行号以及相应的右边常数三部分，右边常数为零的行也不出现。

第五段是变量上下界段，由关键字“UP LINDOBNB”或“LO LINDOBNB”开始，每一行包括：关键字、变量名和上下界值。

最后一段是文件结束符“ENDATA”。

如果把已经生成的MPS格式模型文件调入内存，那么就需要用到下一节的RMPS命令。

§ 7.2 MPS 格式文件读取命令 RMPS

1.命令功能：读取已存盘的MPS格式文件

2.命令格式：

 : RMPS[文件名·扩展名]<CR>

3.命令使用：使用RMPS命令后，屏幕上会显示出模型的一些重要信息，但不显示上一节的TYPE命令下的内容。且看下例：

```
: rmps mps.lis
NAME"AN EXAMPLE OF MPS FORMAT FILE"           ( MAX)
CANDIDATE OBJECTIVE ROW(S) IS(ARE): 1
MAX OR MIN ?
? max
ROWS=  4  VARS=  5  INTEGER VARS=  3( 2 = 0/1)  QCP=  0
NONZEROES= 23  CONSTRAINT NONZ= 15( 2 = +-1)  DENSITY=0.958
SMALLEST AND LARGEST ELEMENTS IN ABSOLUTE VALUE=1.00  19.00
OBJ=MAX, NO. <,=,>: 1 1 1, GUBS <= 1  VUBS >=  0
SINGLE COLS=  0  REDUNDANT COLS=  0
:
```

如果MPS格式文件中有多个目标函数，则当读入文件后将有提示要求用户确定用哪一个目标函数，由于MPS格式文件中不包括极大或极小信息（因为其中可能包含多个目标函数），用户还需要回答优化的方向是极大化还是极小化。

以上屏幕输出还提供了模型的一些主要信息，如变量个数、行个数、非零元素个数，非零元素占有所有元素个数的比例（即非零元素密度）、模型中最大和最小元素的（绝对）值。

MPS格式文件除了有以上优点以外，最主要的是**MPS**格式是线性规划模型通用的文件格式，各种线性规划软件都以**MPS**格式的模型文件为基本文件格式。

第八章 其它命令

除了已介绍过的若干命令之外，LINDO还有一些对话参数命令，这些命令虽非必须，但是使用了它们之后，可以给用户带来极大的方便。它们是：简洁信息输出命令TERS(Terse)。冗长信息输出命令VERB (Verbose)。成批信息输出命令BAT(Batch)和打印机宽度设置命令WIDTH。

§ 8.1 简洁信息输出命令 TERS(Terse)

1.命令功能：使屏幕上显示的运算结果简化。

2.命令格式：

```
: TERS<CR>
```

3.命令使用：在GO命令之前施行TERS命令，屏幕上显示出的解答将只有目标函数值。例如：

```
: LOOK ALL
MAX 2X1+8X2+6X3
SUBJECT TO
2) 8X1+3X2+2X3<=250
3) 2X1+X2<=50
4) 4X1+3X3<=150
END

: TERS

: GO
LP OPTIMUM FOUND AT STEP 2
OBJ VALUE=700.000000

: QUIT
```

对于0—1规划问题，如果用了TERS命令后在用命令GO，则分支定界过程中的一些中间信息就不在屏幕输出中出现，使输出内容比较简洁。请看下例并与第六章中求解0—1规划的例子比较：

```
: LOOK ALL
```

```

MAX 3X1+4X2+5X3+3X4+2X5
SUBJECT TO
2) 2X1+3X2+4X3+5X4+X5<=11
3) 3X1+4X2+2X3+X4+3X5<=16
4) 7X1+2X2+5X3+4X4+2X5<=19
END
INTE          3
: TERS
: GO
LP OPTIMUM FOUND AT STEP 6
NEW INTEGER SOLUTION AT BRANCH 3 PIVOT 10

OBJECTIV EFUNCTION VALUE

1)16.0000000

VARIABLE      VALUE      REDUCED COST
X1             1.000000    -1.000000
X2             1.000000    -2.000000
X3             1.000000    -3.000000
X4             .000000     7.000000
X5             2.000000     .000000

ROW           SLACK      DUALPRICES
2)           .000000     2.000000
3)           1.000000     .000000
4)           1.000000     .000000

NO. ITERATIONS = 10
BRANCHES=3 DETERM.=1.000E0
BEST REMAINING SOLUTION NO BETTER THAN 16.00000
ENUMERATION COMPLETE. BRANCHES=3 PIVOTS = 10

LAST INTEGER SOLUTION IS THE BEST FOUND

```

§ 8.2 冗长信息输出命令 VERB(Verbose)

1.命令功能：该命令的功能就是取消简洁输出命令TERS，恢复冗长信息输出。

2.命令格式：

: VERB<CR>

§ 8.3 成批信息输出命令 BAT (Batch)

在DIVE命令中，所有的屏幕输出都将转到指定的磁盘文件中，但操作过程中的键盘输入信息却不能存入该磁盘文件。BAT命令将每一次键盘输入都作一次相同的键盘输出。因此，在BAT命令以后所作的键盘输入和屏幕输出，都可以转到指定的磁盘文件中。这样就可以使磁盘文件中的内容更完整，更具有可读性。

§ 8.4 打印机宽度设置命令 WIDTH

1.命令功能：设置打印机宽度

2.命令格式：

 : WIDTH <n>

其中n为需要设置的打印机宽度。