راهنمای عملیاتی نرم‌افزار GAMS

GAMS چیست؟

سیستم GAMS یک زبان برنامه‌نویسی مدل‌سازی با قابلیت بالاست.

از GAMS برای حل مسائل برنامه‌ریزی خط (LP)، برنامه‌ریزی غیرخطی (NLP)، برنامه‌ریزی صحیح مختلط (MIP)، برنامه‌ریزی خطی صحیح مختلط (MINLP) و مسائل مکمل خطی (MCP) استفاده می‌کنند.

روش نصب و راه‌اندازی GAMS بر روی کامپیوتر شخصی

GAMS در محیطهای DOS ، WINDOWS و UNIX قابل اجراست. در این جا راه‌اندازی GAMS را برای اجرا تحت DOS بیان می‌کنیم.

توجه کنید که سیستم GAMS تحت DOS شامل ۴ فلاپی ۱/۴۴ می‌باشد.

۱- ابتدا یک شاخه دلخواه روی کامپیوتر شخصی خود (مثلاً GAMS) درست کنید. سپس هر چهار فلاپی را روی شاخهٔ GAMS کپی کنید. (ترتیب کپی کردن فلاپی‌ها مهم نیست.)

۲- برنامهٔ gamsinst را اجرا کنید.

این برنامه تمام فایل‌های فشردهٔ Pck.* را باز می‌کند.

چگونه یک برنامه GAMS بنویسیم

برای حل یک مسأله توسط GAMS اولین مرحله عبارت است از تشکیل یک فایل متن که بیان‌کننده مسأله شما به زبان GAMS است. از ویرایشگر متن مثلا Edit در سیستم عامل DOS یا هر ویرایشگر متنی که در اختیار دارید می‌توانید استفاده کنید. معمولا فایل حاصل را با پسوند GMS مشخص می‌کنیم. مثلا اگر برنامه‌ای به نام TEST می‌نویسید فایلی به نام TEST.GMS ایجاد کنید.

نکاتی در مورد نحوهٔ نوشتن برنامهٔ GAMS

۱- برای رفتن به خط بعدی از ENTER استفاده کنید.

۲- هرگز از کلید TAB استفاده نکنید، زیرا موجب پیغام خطا می‌شود.

۳- هر خطی که با '*' شروع شود به عنوان توضیح در نظر گرفته می‌شود و در برنامه خوانده نمی‌شود.

۴- برای GAMS حروف کوچک و بزرگ فرقی ندارد.

۵- یک برنامهٔ GAMS شامل بخش‌های متعددی است، هر بخش با یک سیمی‌کالون (؛) خاتمه می‌یابد.

یک برنامهٔ نمونهٔ GAMS

قبل از پرداختن به توضیحات مربوط به بخش‌های مختلف یک برنامهٔ GAMS کلیات برنامهٔ زیر را ملاحظه نمائید.

مسأله:

$$\min \quad -x_1 - 2x_2 + x_3 - x_4 - 4x_5 + 2x_6$$

به طوری که

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \le 6$$

$$2x_1 - x_2 - 2x_3 + x_4 \le 4$$

$$x_3 + x_4 + 2x_5 + x_6 \le 4$$

$$x_1, x_2, x_3, x_4, x_5, x_6 \ge 0$$

برنامهٔ مربوط به این مسأله در بخش‌های بعدی به عنوان TEST ارجاع داده خواهد شد.

```
*           OPTION LAMRO = 0 , LIMCOL = 0
SETS
            J / C1*C6 /
            I / B1*B3 / ;
PAPAMETERS
            B ( I )
```

```
                / B1 6
                  B2 4
                  B3 4 /
        C ( J )
                / C1 -1
                  C2 -2
                  C3 1
                  C4 -1
                  C5 -4
                  C6 2 / ;
TABLE
                  A ( I , J )
                  C1      C2      C3      C4      C5      C6
        B1        1       1       1       1       1       1
        B2        2       -1      -2      4       0       0
        B3        0       0       1       1       2       1 ;
VARIABLES
        COSTS
        POSITIVE VARIABLES
                                        X ( J ) ;
EQUATION
        OBJECTIVE
        Y ( I ) ;
OBJECTIVE . .           COSTS = E = SUM ( J , C ( J )  X ( J ) ) ;
Y ( I ) . .             SUM ( J , A ( I , J ) * X ( J ) ) = L = B ( I ) ;
MODEL TEST / ALL / ;
SOLVE TEST USING LP MINIMIZING COSTS ;
DISPLAY     X . L   , X . M ;
```

توضیح

بخش **SETS**

در این بخش تمام اندیس‌هایی که در نوشتن معادلات مدل استفاده می‌شوند معرفی می‌گردند. این اندیس‌ها عباراتی صرفی – عددی هستند. به عنوان مثال در برنامهٔ TEST بخش SETS ، ضرایب تابع هدف توسط J و تعداد قیود توسط I مشخص شده‌اند.

توجه شود که در اصل J را توسط ضرایب تابع هدف به صورت

J / C1 , C2 , C3 , C4 , C5 , C6 /

یا

J / C1 * C6 /

و I را توسط عناصر صمت راست قیود نامساوی به صورت

I / B1 , B2 , B3 /

یا

I / B1 * B3 /

شمرده‌ایم.

توجه: نام هر اندیس تا ده کاراکتر اجازه داده شده و درج فاصله یا جای خالی در نام اندیس اجازه داده نشده است.

بخش **PARAMETERS**

در GAMS پارامترها بردارهای ثابت معلوم و مقدار پارامترها به عنوان داده‌ها در این بخش تعریف می‌شوند. به منظور تعریف مقادیر بردارها، لازم است، اندیس بردارها قبلا در بخش SETS تعریف شده باشد.

در برنامهٔ TEST بردارهای B و C دارای یک مقدار عددی به ترتیب برای هر عنصر مجموعهٔ I و J به صورت زیر است.

PAPAMETERS

```
B ( I )
      / B1 6
        B2 4
        B3 4 /
C ( J )
      / C1 -1
        C2 -2
        C3 1
        C4 -1
        C5 -4
        C6 2 / ;
```

بخش **TABLE**

در GAMS یک ماتریس، مثلا ماتریس ضرایب قیود در LP ، ماتریس ضرایب هزینه در حمل و نقل و... ، در بخش TABLE آورده می‌شود.

به صورت زیر :

```
TABLE
          A ( I , J )
          C1      C2      C3      C4      C5      C6
B1        1       1       1       1       1       1
B2        2       -1      -2      4       0       0
B3        0       0       1       1       2       1 ;
```

توجه شود که تنها یک ماتریس در هر بخش TABLE می‌توان وارد کرد و لازم است با یک سیمی‌کالون (؛) خاتمه یابد. برای وارد کردن چندین ماتریس، همان تعداد TABLE لازم است.

بخش متغیرها **VARIABLES**

در بخش VARIABLES مجهولات مسأله را لیست می‌کنیم. متغیرها می‌توانند اسکالر باشند، یک بردار باشند، یک ماتریس دوبعدی باشند، یک ماتریس سه‌بعدی باشند و غیره ...

در GAMS متغیرهای نامنفی با عنوان POSITIVE VARIABLES و متغیرهای نامثبت با عنوان NEGATIVE VARIABLES معرفی می‌شوند.

متغیرها می‌توانند BINARY باشند که در این صورت تنها مقادیر صفر و یک می‌گیرند.

بخش معادلات **EQUATIONS**

بخش بعدی بخش معادلات است که اسامی تابع هدف و قیود دیگر مسأله در آن تعریف می‌شود. نامعادلات شبیه معادلات در این بخش ظاهر می‌شوند.

در برنامهٔ TEST نام OBJECTIVE برای عبارتی که باید می‌نیمم گردد (تابع هدف) انتخاب شده است. نام Y(I) برای قیود مسأله انتخاب شده است.

بعد از نام‌گذاری، معادلات وارد می‌شوند. ابتدا نام معادله، به دنبال آن دو نقطه (..) آن گاه تعریف جبری قید که خیلی شبیه علائم استاندارد ریاضی است، می‌آید. ضرب به وسیلهٔ علامت (*) ، تقسیم به وسیلهٔ علامت اِسلش (/)، توان به وسیلهٔ (**) تعریف می‌شود.

بعضی از توابع استاندارد

نام	شرح	تعریف ریاضی
ABS	قدر مطلق	|arg|
COS	کسینوس	COS (arg)
EXP	تابع نمایی	exp (arg)

لگاریتم در مبنای e از arg	لگاریتم طبیعی	LOG
لگاریتم در مبنای ۱۰ از arg	لگاریتم معمولی	LOG10
Qrg*arg	توان دوم	SQR
Sqrt(arg)	ریشه دوم	SQRT

بعضی از علائم ریاضی

علائم ریاضی معمولی	GAMS
$\sum_{i \in I}$	SUM (I ,)
$\sum_{i \in I} \sum_{j \in J}$	SUM ((I , J) ,)
=	= E =
≥	= G =
≤	= L =

در GAMS راهی برای بیان کران بالا، کران پایین و قیود تساوی که با یک متغیر سر و کار دارند وجود دارد، مثال زیر را ببینید:

علائم استاندارد ریاضی	GAMS
$x \leq ۲۰$	X = L = 20
$x \geq ۲۰$	X = G = 20
$x = ۲۰$	X = E = 20

به جای تعاریف فوق قبل از EQUATION، دستورات زیر را می‌توانیم بیاوریم.

```
X . UP = 20 ;
X . LO = 20 ;
```

X . FX = 20 ;

که در آن UP ، LO ، FX بیان‌کننده کران بالا، کران پایین و مقدار ثابت هستند.

بخش **MODEL** و **SOLVE**

در آخرین قسمت برنامه، MODEL داده می‌شود، نام مدل (نام فایل) حداکثر ۸ تا ۱۰ کاراکتر بسته به GAMS مورد استفاده، انتخاب می‌شود.

در GAMS کلمهٔ مدل مجموعه‌ای از معادلات است که معمولاً یکی از آنها تابع هدف مسأله است، به عنوان مثال:

MODEL TEST / ALL / ;

این معنی را می‌دهد که تمام معادلاتی را که قبلا معرفی شده است به عنوان مدل TEST در نظر گرفته شود.

از این قسمت می‌توان برای حل تجزیهٔ LP هم استفاده کرد (Decomposition) طوری که علاوه بر تابع هدف قیودی را که برای حل کردن مد نظر است در این قسمت می‌نویسیم.

آخرین خط برنامه:

SOLVE TEST USING LP MINIMIZING COSTS ;

به کامپیوتر اعلام می‌کند که مدل TEST را با استفاده از برنامه‌ریزی خطی (LP) که در کتابخانه برنامه حاضر است، به منظرو می‌نیمم‌سازی متغیر COSTS (تابع هدف) حل کن.

در انتهای برنامه یک خط دلخواه را می‌توان اضافه کرد:

DISPLAY X . L , X . M ;

که این خط باعث می‌شود در انتهای جواب برنامه، مقدار متغیرهای مسأله اولیه و مسألهٔ دوگان به طور خلاصه و جمع و جور نوشته شود.

چگونه یک برنامهٔ GAMS را اجرا کنیم؟

بعد از نوشتن برنامه (در یک ویرایشگر)، از آن ویرایشگر خارج می‌شویم، وقتی وارد صفحهٔ DOS شدیم، دستور زیر را برای اجرای برنامهٔ TEST.GMS می‌نویسیم:

> GAME TEST . GMS (یا ساده‌تر GAMS TEST)

پس از اجرای برنامه، گزارشی از جواب در فایلی به نام TEST . LST ایجاد خواهد شد.به کمک ویرایشگرتان (مثلا EDIT) می‌توانید این فایل را ببینید یا آن را چاپ کنید.

اولین بار که برنامه‌ای را اجرا می‌کنید، احتمالاً شامل خطاهایی خواهد بود، که این خطاها به وسیلهٔ چهار ستاره (****) در سمت چپ خط خطا علامت‌گذاری می‌شود. در جلوی (****)، علامت $ را با یک عدد ملاحظه خواهید کرد، در انتهای برنامه و جلوی همان عدد نوع خطایی را که مرتکب شده‌اید، نوشته شده است، نوع خطا را مطالعه کرده و نسبت به رفع آن اقدام کنید.

بعد از رفع خطاهای برنامه، سرانجام خلاصه‌ای از جواب را در TEST . LST ملاحظه خواهید کرد، که شامل اطلاعاتی است که نیاز خواهید داشت. این قسمت با چهار خط تیره "----" شروع می‌شود. به عنوان مثال خلاصه‌ای از جواب برنامهٔ TEST عبارت است از:

---- EQU Y

	LOWER	LEVEL	UPPER	MARGINAL
B1	6.000	6.000	6.000	-2.000
B2	4.000	4.000	4.000	ESP
B3	4.000	4.000	4.000	-1.000

LOWER LEVEL UPPER MARGINAL

---- VAR Z -INF -16.000 + INF

---- VAR Z

	LOWER	LEVEL	UPPER	MARGINAL
C1	-	-	+ INF	1.000
C2	-	4.000	+ INF	-
C3	-	-	+ INF	4.000
C4	-	-	+ INF	2.000
C5	-	2.000	+ INF	-
C6	-	-	+ INF	5.000
C7	-	-	+ INF	2.000
C8	-	8.000	+ INF	-
C9	-	-	+ INF	1.000

**** REPORT SUMMARY : 0 NONOPT
0 INFEASIBEL
0 UNBOUNDED

---- 36 VARIABLE X . L
C2 4.000, C5 2.000, C8 8.000

---- 36 VARIABLE X . M
C1 1.000, C3 4.000, C4 2.000, C6 5.000, C7 2.000, C9 1.000

اولین قسمت خلاصه جواب شامل معادلات است. قسمت دوم خلاصه جواب مقدار بهین تابع هدف را ارائه می‌دهد که برابر ۱۶- است و در ادامه مقدار بهین متغیره هستند که تحت عنوان LEVEL لیست می‌شوند. در قسمت پنجم و ششم، X.L و X.M را ملاحظه خواهید کرد که همان مقادیر بهینهٔ به ترتیب متغیرهای اولیه و دوگان مسأله می‌باشند.

به توضیح REPORT SUMMARY در قسمت «نکات مهم» دقت شود.

نکات مهم

۱- در TEST.LST بعد از صورت برنامه، خلاصه‌ای از برنامه می‌آید که تمام اطلاعاتی که شما نیاز دارید در آن نوشته شده است. یکی از قسمت‌های مهم آن، قسمتی است به نام MARGINAL که همان مقدار متغیرهای دوگان متناظر با قیود می‌باشد.

۲- در جواب برنامه (در TEST.LST)، قسمتی را ملاحظه خواهید کرد به نام REPORT SUMMARY ، که در این قسمت سه کلمهٔ NONOPT، INFEASIEL ، UNBOUNDED را مشاهده خواهید کرد، اگر در جلوی هر کدام از آن‌ها عدد ۱ نوشته شده باشد یعنی همان حالت اتفاق افتاده است، مثلاً اگر در کنار UNBOUNDED عدد ۱ نوشته شده باشد یعنی مسأله اولیه بی‌کران است.

نوشته شدن عدد صفر طبیعی است.

۳- به نویسنده برنامه GAMS پیشنهاد می‌شود که قبل از نوشتن برنامه، مسأله را به صورت استاندارد درآورده و سپس برنامهٔ GAMS را بنویسد، حسن این عمل در آن است که در قیود مسأله فقط E = = به کار خواهد رفت و خیلی راحت صورت برنامه قابل تعویض برای مسائل دیگر است.

برای آشنایی بیش‌تر یک برنامهٔ مسألهٔ متغیرهای کران‌دار را در مثال زیر می‌آوریم:

$$\max \quad ۲x_۱ + ۶x_۲ - x_۳ - ۴x_۴ + x_۵$$

به طوری که

$$۲x_۱ + x_۲ + ۴x_۳ + x_۴ + x_۵ = ۱۰$$

$$۳x_۱ + ۸x_۲ - ۳x_۳ + x_۴ = ۷$$

$$۰ \le x_۱ \le ۳$$

$$۱ \le x_۲ \le ۴$$

$$۰ \le x_۳ \le ۸$$

$۱ \leq x_{۴} \leq ۲$

$۰ \leq x_{۵} \leq ۲۰$

برنامهٔ GAMS:

```
SETS
        J / C1 * C5 /
        I / B1 * B2 / ;
PARAMETERS
        B ( I )
                / B1    10
                B2      -7 /
        C ( J )
                / C1    2
                C2      6
                C3      -1
                C4      -4
                C5      1 / ;
TABLE
        A ( I , J )
                C1      C2      C3      C4      C5
        B1      2       1       4       1       1
        B2      -3      -8      3       -1      0 ;
VARIABLES
                Z
                POSITIVE VARIABLES
                                        X ( J ) ;
X . LO ( ' C1 ' ) = 0 ;
X . UP ( ' C1 ' ) = 3 ;
X . LO ( ' C2 ' ) = 1 ;
X . UP ( ' C2 ' ) = 4 ;
X . LO ( ' C3 ' ) = 0 ;
X . UP ( ' C3 ' ) = 8 ;
```

```
X . LO ( ' C4 ' ) = 1 ;
X . UP ( ' C4 ' ) = 2 ;
X . LO ( ' C5 ' ) = 0 ;
X . UP ( ' C5 ' ) = 20 ;
EQUATION
        OBJECTIVE
        Y ( I ) ;
OBJECTIVE . .       Z = E = SUM ( J , C ( J ) * X ( J ) ) ;
Y ( I ) . .         SUM ( J , A ( I , J ) * X ( J ) ) = E = B ( I ) ;
MODEL TEST1 USING LP MINIMIZING Z ;
DISPLAY X . L ,     X . M ;
```

برای کامل شدن نمونه‌ها ، برنامهٔ یک مسأله حمل و نقل زیر را می‌نویسیم:

صورت مسأله:

ماتریس c_{ij}

	۱	۲	۳	۴	S_i
۱	۶	۲	-۱	۰	۵
۲	۴	۲	۲	۳	۲۵
۳	۳	۱	۲	۱	۲۵
d_j	۱۰	۱۰	۲۰	۱۵	

برنامهٔ GAMS

```
SETS
        J / D1 , D2 , D3 /
        I / S1 , S2 , S3 , S4 / ;
PARAMETERS
        A ( I )
                / D1    5
                D2      25
                D3      25 /
        B ( J )
```

```
                / S1    10
                S2      10
                S3      20
                S4      15 / ;
TABLE
        C ( I , J)
                S1      S2      S3      S4
        D1      6       2       -1      0
        D2      4       2       2       3
        D3      3       1       2       1 ;
VARIABLES
                Z
                POSITIVE VARIABLES
                                        X ( I , J ) ;
EQUATION
                OBJECTIVE
                DBAL ( I )
                SBAL ( J ) ;
OBJECTIVE . .           Z = E = SUM ( ( I , J ) , C ( I , J ) * X ( I , J ) ) ;
DBAL ( I ) . .          SUM ( J , X ( I , J ) ) = L = A ( I ) ;
SBAL ( J ) . .          SUM ( I , X ( I , J ) ) = G = B ( J ) ;
MODEL TEST2 / ALL / ;
SOLVE TEST2 USING LP MINIMIZING Z ;
```

برنامه‌های کاربردی درس تحقیق در عملیات

برنامه حل مسائل برنامه‌ریزی خطی استاندارد شده به روش سیمپلکس (زبان برنامه‌نویسی Pascal)

```
PROGRAM SIMPLEX;
USES CRT;
TYPE
```

تعریف ساختار آرایه‌های جهتاستفاده در برنامه اصلی و توابع وابسته به آنها

```
        TYP1=ARRAY[1..3]        OF CHAR;
        TYP2=ARRAY[1..10]       OF INTEGER;
```

```
        TYP3=ARRAY[1..10]          OF REAL;
        TYP4=ARRAY[1..10][1..10] OF CHAR;
VAR
        OUT:TEXT;                          تعریف متغیرهای مورد نیاز در طول برنامه

        SW:CHAR;
        P:TYP1;
        N1,N2,I,J,V,W,CO,S,K:INTEGEP;
        A3,HELP,HELP2:REAL;
                                                    A3 مقدار تابع هدف

        A1,A2,C:TYP3;
                                    A1 سطر هدف متغیرهای حقیقی، A2 سطر هدف متغیرهای تصنعی

        B1,B2:TYP4;
                                    B1 ماتریس متغیرهای حقیقی، B2 ماتریس متغیرهای تصنعی

        HELP11,HELP12:TYPE2;
                                    بقیه متغیرهای برنامه همگی کمکی می‌باشند.

FUNCTION RASM : INTEGER ;
```

این تابع ورودی‌هایی را جهت ترسیم یک مرحله از جدول سیمپلکس نیاز دارد می‌گیرد و جدول مربوط به آن مرحله را ترسیم میرکند.

```
VAR
        U:INTEGER;
        T:CHAR;                                  تعریف متغیرهای محلی مورد نیاز

BEGIN
        CLRSCR;
        CO:=CO+1;
        GOTOXY(1,10);
        WRITE('پایه',P[3]);
```

آرایه P در برنامه اصلی تعریف شده است.

```
WRTTE(OUT,'پایه',P[3]);
```

این قسمت جهت رسم همزمان جدول در یک فایل خروجی (که در برنامه اصلی معرفی شده است) نوشته شده است.

```
FOR I:=1 TON 1 DO
        BEGIN
                WRITE('  X',I,'  ',);
                WRITE(OUT,'  X',I,'  ');
```

از این به بعد تمام دستورات WRITE که در اول آنها عبارت OUT آمده است جهت نوشتن در فایل خروجی بکار می‌روند.

```
        END;
WRITE('  '.P[3]);
WRITE(OUT,'  '.P[3]);
FOR I:=1 TO N2 DO
        BEGIN
                WRITE('  S',I,'  ');
                WRITE(OUT,'  S',I,'  ');
        END;
WRITELN(P[3],'جواب');

WRITELN(OUT,P[3],'جواب');

FOR I:=1 TO 6 DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITE(P[2]);
WRITE(OUT,P[2]);
FOR I:=1 TO N1*10-(N1-2) DO
        BEGIN
                WRITE(P[1]);
```

```
                WRITE(OUT,P[1]);
        END;
WRITE(P[2]);
WRITE(OUT,P[2]);
FOR I:=1 TO (N2*8)+3 DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITE(P[2]);
WRITE(OUT,P[2]);
FOR I:=1 TO 9 DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITELN;
WRITELN(OUT);
WRITE('XO   ',P[3]);
WRITE(OUT,'XO   ',P[3]);
FOR I:=1 TO N1 DO
        BEGIN
                WRITE('  ',-A1[I]:7:2,'   ');
                WRITE(OUT,'   ',-A1[I]:7:2,'   ');
        END;
WRITE('   ',P[3]);
WRITE(OUT,'   ',P[3]);
FOR I:=1 TO N2 DO
        BEGIN
                WRITE(A2[I]:7:2,'   ');
                WRITE(OUT,A2[I]:7:2,'   ');
        END;
WRITE('   ',P[3],A3:7:2);
WRITE(OUT,'   ',P[3],A3:7:2);
```

```
FOR I:=1 TO 6 DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITE(P[2]);
WRITE(OUT,P[2]);
FOR I:=1 TO N1*10-(N1-2) DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITE(P[2]);
WRITE(OUT,P[2]);
FOR I:=1 TO (N2*8)+3 DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITE(P[2]);
WRITE(OUT,P[2]);
FOR I:=1 TO 9 DO
        BEGIN
                WRITE(P[1]);
                WRITE(OUT,P[1]);
        END;
WRITELN;
WRITELN(OUT);
HELP11[CO]:V;
HELP12[CO]:W;
FOR I:=1 TO N2 DO
        IF I=W THEN
                BEGIN
                        T:='X';
```

```
                    U:=V;
                END
            ELSE IF (CO<>0) AND (I=HELP12[CO-1]) THEN
                BEGIN
                    T:='X';
                    U:=HELP11[CO-1];
                END
            ELSE
                BEGIN
                    T:='S';
                    U:=I;
                END;
            WRITE(T,U,'   ',P[3]);
            WRITE(OUT,T,U,'   ',P[3]);
            FOR J:=1 TO N1 DO
                BEGIN
                    WRITE('   ',B1[I,J]:7:2);
                    WRITE(OUT,'   ',B1[I,J]:7:2);
                END;
            FOR S:=1 TO N1-2 DO
                BEGIN
                    WRITE('   ');
                    WRITE(OUT,'   ');
                END;
            WRITE('   ',P[3]);
            WRITE(OUT,'   ',P[3]);
            FOR K:=1 TO N2 DO
                BEGIN
                    WRITE(B2[I,J]:7:2,'   ');
                    WRITE(OUT,B2[I,J]:7:2,'   ');
                END;
            WRITELN('   ',P[3],'   ',C[I]:7:2);
            WRITELN(OUT,'   ',P[3],'   ',C[I]:7:2);
END
```

```
        WRITELN;
        WRITELN(OUT);
END;                                                    پایان تابع مربوط به رسم

BEGIN
        CLRSCR;
        CO:=-1
        ASSIGN(OUT,'OUTPUT.TXT');
```

در این قسمت فایل خروجی با نام OUTPUT.TXT ساخته میشود.

```
REWRITE(OUT);
P[1]:='-';
```

تعریف آرایه P

```
P[2]:='+';
P[3]:='|';
WRITE ('تعداد متغیرهای مسأله را وارد کنید NUMBER_VAR=');

READLN(N1);                                             خواندن تعداد متغیرها

WRITELN(OUT,'تعداد متغیرهای مسأله را وارد کنید NUMBER_VAR=',N1);

WRITE('ضرایب تابع هدف را وارد کنید C[I]=');

FOR I:=1 TO N1 DO
        READ(A1[I]);
```

خواندن ضرایب تابع هدف و اختصاص آرایه A1 به آنها

```
READLN;                                                 انتقال مکان نما به اول خط بعد

WRITE('ضرایب تابع هدف را وارد کنید C[I]=');

FOR I:=1 TO N1 DO
        WRITE(OUT,A1[I]:7:2);                   نوشتن ضرایب وارد شده در فایل خروجی

WRITELN(OUT);
WRITE ('تعداد قیود مسأله را وارد نمایید NUMBER_GHEYD=');
```

```
READLN(N2);                                          خواندن تعداد قیود

WRITE(OUT,'تعداد قیود مساله را وارد نمائید NUMBER_GHEYD=',N2);

FOR I:=1 TO N2 DO
        BEGIN
                WRITE('ضرایب قید',I,'را وارد کنید :');

                FOR J:=1 TO N1 DO
                        READ(B1[I,J]);              خواندن ضرایب قیود به ترتیب شماره

                READLN;
        END;
FOR I:=1 TO N2 DO
        BEGIN
                WRITE(OUT,'ضرایب قید',I'را وارد کنید :');

                FOR J:=1 TO N1 DO
                        WRITE(OUT,B1[I,J]:7:2);
                WRITELN(OUT);
        END;
FOR I:=1 TO N2 DO
        BEGIN
                WRITE('جواب قید',I,'را وارد کنید :');

                READLN(C[I]);                       خواندن جواب قیود به ترتیب شماره

                WRITE('جواب قید',I,'را وارد کنید :',C[I]:7:2);

        END;
WRITELN(OUT);
WRITELN(OUT);
FOR I:=1 TO N2 DO
        FOR J:=1 TO N2 DO
                IF I=J THEN
                        B2[I,J]:=1
```

```
                ELSE
                        B2[I,J]:=0;
A3:0;
CLRSCR;
V:=0;
```

V را متناظر با اندیس متغیر وارد شونده در نظر میگیریم

```
W:=-1;
```

W را متناظر با اندیس متغیر خارج شونده در نظر میگیریم

```
RASM;
```

رسم اولین جدول سیمپلکس

```
WHILE TURE DO
```

در اینجا مراحل زیر تا رسیدن به یک جواب بهینه ادامه می یابد

```
        BEGIN
                FOR I:=1 TO N1 DO
                        IF A1[I]>A1[V] THEN
```

تعیین اندیس متغیر واردشونده

```
                                V:=I;
                IF –A1[V]>=0 THEN
```

در صورتی که مقدار متغیر واردشونده مثبت باشد

```
                        BEGIN
```

به جواب بهینه رسیده ایم

```
                                WRITELN(A3:7:2,':جواب بهینه مسأله برابر است با');
                                WRITELN(OUT,A3:7:2,':جواب بهینه مساله برابر است با');
                                IF SW='Y' THEN
```

در صورتی که مقدار SW برابر Y باشد جواب تباهیده است (در زیر بیشتر توضیح داده میشود)

```
                                        BEGIN
                                                WRITELN('جواب مساله تباهیده است');
                                                WRITELN(OUT,'جواب مساله تباهیده است');
                                        END;
                                READKEY;
```

مکث جهت رویت جواب

```
                CLSE(OUT);                          بستن فایل خروجی

                HALT;                               خروج از برنامه

        END;
WRITELN;
WRITELN(OUT);
WRITELN('متغیر واردشونده =',V);

WRITELN(OUT,'متغیر وارد شونده =',V);

FOR J:=1 TO N2 DO                                   تعیین اندیس متغیر خارج شونده

        IF B1[J,V]>0 THEN
                IF C[J]/B1[J,V]<=C[W]/B1[W,V]THEN
                        W:=J
WRITELN('متغیر خارج شونده = S',W);

WRITELN(OUT,'متغیر خارج شونده =S',W);

WRITELN;
WRITELN(OUT);
HELP2:=B1[W,V];                                     صفر کردن ستون مربوط به واردشونده

FOR I:=1 TO N1 DO
        B1[W,I]:=B1[W,I]/HELP2;
FOR J:=1 TO N2 DO
        B2[W,J]:=B2[W,J]/HELP2;
C[W]:=C[W]/HELP2;
I:=1
WHILE I<=N2 DO
        BEGIN                                       بهینه کردن جدول سیمپلکس

                IF I=W THEN
                        I:=I+1;
                HELP:=-B1[I,V];
```

```
                    FOR J:=1 TO N1 DO
                        B1[I,J]:=B1[W,J]*HELP+B1[I,J];
                    FOR J:=1 TO N2 DO
                        B2[I,J]:=B2[W,J]*HELP+B2[I,J];
                    C[I]:=C[W]*HELP+C[I];
                    IF C[W]=0 THEN
                        SW:='Y'
                    I:=I+1
                END;
            HELP:=A1[V];
            FOR J:=1 TO N1 DO
                A1[J]:=-(B1[W,J]*HELP-A1[J]0;
            FOR J:=1 TO N2 DO
                A2[J]:=-(B2[W,J]*HELP-A2[J]0;
            A3:=C[W]*HELP+A3                    بهینه کردن مقدار تابع هدف

            READKEY;                            خواندن کلید جهت رویت تغییرات

            RASM;                               رسم جدول بعدی با اطلاعات بهینه شده

        END;
CLOSE(OUT);                                     بستن فایل خروجی

READKEY;                                        خواندن کلید جهت رویت تغییرات

END.                                            پایان برنامه
```

چند مثال

تعداد متغیرهای مسأله را وارد کنید NUMBER_VAR = 3

ضرایب تابع هدف را وارد کنید C[I] = 1.00 2.00 3.00

تعداد قیود مسأله را وارد نمائید NUMBER_GHEYD = 3

ضرایب قید ۱ را وارد کنید : 5.00 7.00 0.00

ضرایب قید ۲ را وارد کنید : 1.00 -2.00 5.00

ضرایب قید ۳ را وارد کنید : 1.00 -2.00 -3.00

جواب قید ۱ را وارد کنید : 1.00

جواب قید ۲ را وارد کنید : -2.00

جواب قید ۳ را وارد کنید : 7.00

پایه	X1	X2	X3	S1	S2	S3	جواب
X0	-1.00	-2.00	-3.00	0.00	0.00	0.00	0.00
S1	5.00	7.00	0.00	1.00	0.00	0.00	1
S2	1.00	-2.00	5.00	0.00	1.00	0.00	-2.00
S3	1.00	-2.00	-3.00	0.00	0.00	1.00	7.00

متغیر وارد شونده = X3

متغیر خارج شونده = S2

پایه	X1	X2	X3	S1	S2	S3	جواب
X0	-0.40	-3.20	0.00	0.00	0.60	0.00	-1.20
S1	5.00	7.00	0.00	1.00	0.00	0.00	1
X3	0.20	-0.40	1.00	0.00	0.20	0.00	-0.40
S3	1.60	-3.20	0.00	0.00	0.60	1.00	5.80

متغیر وارد شونده = X2

متغیر خارج شونده = S1

پایه	X1	X2	X3	S1	S2	S3	جواب
X0	1.89	0.00	0.00	0.46	0.60	0.00	-0.74
X2	0.71	1.00	0.00	0.14	0.00	0.00	0.14
X3	0.49	0.00	1.00	0.06	0.20	0.00	-0.34
S3	3.89	0.00	0.00	0.46	0.60	1.00	6.29

جواب بهینه مسأله برابر است با : 0.47-

تعداد متغیرهای مسأله را وارد کنید NUMBER_VAR = 4

ضرایب تابع هدف را وارد کنید C[I] = 3.00 2.00 5.00 1.00

تعداد قیود مسأله را وارد نمائید NUMBER_GHEYD = 2

ضرایب قید ۱ را وارد کنید : 8.00 7.00 1.00 0.00

ضرایب قید ۲ را وارد کنید : 1.00 2.00 -1.00 5.00

جواب قید ۱ را وارد کنید : 2.00

جواب قید ۲ را وارد کنید : 5.00

پایه	X1	X2	X3	X4	S1	S2	جواب
X0	-3.00	-2.00	-5.00	-1.00	0.00	0.00	0.00
S1	8.00	7.00	1.00	0.00	1.00	0.00	2.00
S2	1.00	2.00	-1.00	5.00	0.00	1.00	5.00

متغیر وارد شونده = X3

متغیر خارج شونده = S1

پایه	X1	X2	X3	X4	S1	S2	جواب
X0	37.00	33.00	0.00	-1.00	5.00	0.00	10.00
X3	8.00	7.00	1.00	0.00	1.00	0.00	2.00
S2	9.00	9.00	0.00	5.00	1.00	1.00	7.00

X4 = متغیر وارد شونده

S2 = متغیر خارج شونده

پایه	X1	X2	X3	X4	S1	S2	جواب
X0	38.80	34.80	0.00	0.00	5.20	0.20	11.40
X3	8.00	7.00	1.00	0.00	1.00	0.00	2.00
X4	1.80	1.80	0.00	1.00	0.020	0.20	1.40

جواب بهینه مسأله برابر است با : 11.40

تعداد متغیرهای مسأله را وارد کنید NUMBER_VAR = 2

ضرایب تابع هدف را وارد کنید C[I] = 1.00 2.00

تعداد قیود مسأله را وارد نمائید NUMBER_GHEYD = 4

ضرایب قید ۱ را وارد کنید : 2.00 0.00

ضرایب قید ۲ را وارد کنید : -9.00 1.00

ضرایب قید ۳ را وارد کنید : 2.00 3.00

ضرایب قید ۴ را وارد کنید : 2.00 2.00

جواب قید ۱ را وارد کنید : 1.00

جواب قید ۲ را وارد کنید : 2.00

جواب قید ۳ را وارد کنید : 3.00

جواب قید ۴ را وارد کنید : 4.00

پایه	X1	X2	S1	S2	S3	S4	جواب
X0	-1.00	-2.00	1.00	0.00	0.00	0.00	0.00
S1	2.00	0.00	1.00	0.00	0.00	0.00	1.00
S2	-9.00	1.00	0.00	1.00	0.00	0.00	2.00
S3	2.00	2.00	0.00	0.00	0.00	1.00	4.00
S4	2.00	2.00	0.00	0.00	0.00	1.00	4.00

متغیر وارد شونده = X2

متغیر خارج شونده = S3

پایه	X1	X2	S1	S2	S3	S4	جواب
X0	0.33	0.00	0.00	0.00	0.67	0.00	2.00
S1	2.00	0.00	1.00	0.00	0.00	0.00	1.00

S2	-9.67	0.00	0.00	1.00	-0.33	0.00	1.00
X2	0.67	1.00	0.00	0.00	0.33	0.00	1.00
S4	0.67	0.00	0.00	0.00	-0.67	1.00	2.00

جواب بهینه مسأله برابر است با : 2.00