```
SETS:
! Gymnast selection problem.
 How best to select which candidates are part of a team,
 and to which specialty each candidate is assigned.
 A particular application is to selecting a gymnastics team;
!Refs:
    Ellis, P. and R. Corn (1984), "Using bivalent integer programming to select
  teams for intercollegiate women's gymnastics competition," Interfaces, 14(3):41-46.
    Eilon, S. and  A. Land, (1986) "Note: Further Gymnastics," Interfaces, 16(2):69-71.
    Land, A. and S. Powell (1985), "More gymnastics," Interfaces, 15(4):52-54;
! Keywords: Assignment, Corn, Eilon, Ellis, Gymnastics, Land, LINGO, Powell, Sports, Team;
  Gymnast: Y;
  Event;
  GxE( Gymnast, Event): S, X, Z;
Endsets
Data:
!CaseAll;    MAKINT = 1;   ! = 1 if make variables integer, else 0;
!CaseAll;    NPEVENT = 6;  ! Number gymnasts assigned to each event;
!CaseAll;    NALLRND = 4;  ! Number of all rounders required;
!CaseAll;    NCTOP = 5;    ! Count only the NCTOP highest scores;
!CaseAll;    Gymnast = NADIA1 BRUCE2 SIMONE3 DOUG4 MARYLU5 LINUS6 OLGA7 LIDA8 NASTIA9
LUCY10;
!CaseAll;    Event =
     VAULT BARS BEAM FLOOR;
! Case1
  S = 9.3   9.3  9.2  9.5
      9.1   9.0  8.8  8.7
      9.2   9.0  9.2  9.0
      9.0   8.6  8.6  8.65
      8.8   8.7  8.5  8.5
      8.7   7.5  7.5  8.7
      8.5   8.8  8.7  8.5
      9.1   9.0  9.0  9.2
      9.2   7.0  7.0  9.1
      8.9   8.9  9.1  8.7
;
! Case2
  S = 9.5   9.5  9.5  9.5
      9.5   9.5  9.5  9.5
      9.2   9.2  8.6  8.6
      8.6   8.6  9.2  9.2
      9.0   9.1  8.6  8.6
      9.1   7.5  7.5  8.7
      8.6   8.6  9.1  9.1
      9.2   9.2  8.6  8.6
      8.6   8.6  9.2  9.2
      8.9   8.9  8.9  8.9
;
! Case3;
  S = 9.5   9.5  8.6  8.6
      8.6   9.5  9.5  8.6
      9.5   8.6  9.5  8.6
      8.6   8.6  9.5  9.5
      9.4   9.1  9.0  9.4
      9.2   9.0  9.4  9.1
      9.4   9.1  9.4  9.0
      9.3   9.2  9.2  9.3
      9.2   9.3  9.2  9.3
      9.2   9.3  9.2  9.3
;
! To take data from an Excel spreadsheet,
   1) replace a statement like MAKINT = 0, by MAKINT = @OLE(),
   2) make sure the spreadsheet has range names matching the LINGO names,
   3) make sure the only spreadsheet open is the desired one;

!CaseXcl     MAKINT = @Ole();   ! = 1 if make variables integer, else 0;
!CaseXcl     NPEVENT = @Ole();  ! Number gymnasts assigned to each event;
!CaseXcl     NALLRND = @Ole();  ! Number of all rounders required;
```

```
!CaseXcl     NCTOP = @Ole();    ! Count only the NCTOP highest scores;
!CaseXcl     Gymnast = @Ole();  ! Names of candidates;
!CaseXcl     Event = @Ole();    ! Names of events;
!CaseXcl     S = @OLE(, SCORE); ! Score matrix;
Enddata

Submodel HuDuzWat:
! Parameters:
    S( i, j) = expected score of gymnast i in event j;
! Variables:
    Y( i) = 1 if gymnast i is selected as all-rounder,
    X( i, j) = 1 if i participates in event j,
    Z( i, j) = 1 if score of gymnast i is counted for event j
     ;
  NEVENTS = @SIZE( EVENT); ! Number of events;
! Maximize expected score of team;
    MAX = @SUM( GxE( i, j): S( i, j) * Z( i, j));
! If Gymnast i selected as all-rounder, must participate in every event j;
   @For( GxE( i, j):
      X( i, j) >= Y( i);
       );

  @For( EVENT( j):
     @SUM( GYMNAST( i): Z( i, j)) <= NCTOP;  ! Count only the NCTOP highest scores;
      );

  @SUM( GYMNAST( i): Y( i)) >= NALLRND;  ! Must have enough All-rounders;

  @FOR( EVENT( j):
 ! Must have NPEVENT participants in each event;
    @Sum( GYMNAST( i): X( i, j)) = NPEVENT;
      );

   @For( GxE( i, j):
       Z( i, j) <= X( i, j);  ! Score of i counted implies i participated in event;
       @BND( 0, X( i, j), 1); ! Cannot count a gymnast more than once;
       );
! If MAKINT = 1, then make all X variables binary;
   @For( GxE( i, j) | MAKINT:
       @Bin( X( i, j));  ! (Perhaps) make X variables binary integer;
       );
Endsubmodel
Calc:
  @Solve( HuDuzWat);

! Display a simple report;
  @For( GYMNAST( i) | Y( i) #GT# 0:
     @Write( ' Gymnast ', @Format( GYMNAST( i),'8s'),' is an all-rounder     (',Y( i),')',
@NEWLINE( 1));
      );
  @Write( @NEWLINE( 1));
  @For( GxE( i, j) | X( i, j) #GT# 0:
     @Write( ' Gymnast ', @Format( GYMNAST( i),'8s'),' participates in ', Event( j),'
(', X( i, j),')', @NEWLINE( 1));
      );
Endcalc
```