```
# IP model of Consultant Hiring problem on AMPL

set I;  # set of projects
param profit{i in I}; #profits from projects excluding consulting costs

set J;  # set of consultants
param cost{j in J};  # weekly cost of consultant j

set ATTR;  # set of K attributes

param cons_attr{j in J, k in ATTR};
# is 1 if consultant j possesses attribute k, 0 otherwise

param proj_attr{i in I, k in ATTR};
# number of consultants needed to possess attribute k in project i

param b{i in I};  # first week of project i
param e{i in I};  # last week of project i

set WEEKS:= min{i in I}b[i]..max{i in I}e[i];
# the range of the weeks when the projects are planned

param up{w in WEEKS, i in I}:= if b[i]<=w<=e[i] then 1 else 0;
#  this parameter is 1 if project i is planned to be up in week w

var proj{i in I} binary;   #  is 1 if project i is pursued

var assign{j in J, i in I} binary;   #  is 1 if consultant j is assigned to project i

maximize Profit:                    # profit from projects – consultant costs
        sum{i in I}profit[i]*proj[i]
        - sum{j in J, i in I}cost[j]*(e[i]-b[i]+1)*assign[j,i];

subject to Number_of_Consultants{i in I, k in ATTR}:
        sum{j in J}cons_attr[j,k]*assign[j,i] >= proj_attr[i,k]*proj[i];
# number of consultants possessing attribute k in project i should be at least proj_attr[i,k]
# if project i is pursued

subject to one_project_at_a_time{j in J, w in WEEKS}:
                sum{i in I}up[w,i]*assign[j,i] <= 1 ;
# consultant j can work only on one project in any week w

subject to only_pursued_projects{i in I, j in J}:
                assign[j,i] <= proj[i];
# consultants can work only on projects that are pursued
```

```
data;

set I:= A B C D;

param profit[*] :=
A       250
B       300
C       200
D       400 ;

set J:= Tom Jim Ann Tony;

param cost[*] :=
Tom    25
Jim    36
Ann    28
Tony   30 ;

set ATTR:= C++       IP       Linux;

param cons_attr[*,*]
:       C++    IP       Linux  :=
Tom     1      0        1
Jim     1      1        0
Ann     0      1        1
Tony    1      0        1 ;

param proj_attr[*,*]
:       C++    IP       Linux  :=
A       2      1        2
B       1      1        1
C       2      0        1
D       2      1        0 ;

param b[*] :=
A       2
B       1
C       3
D       2;
param e[*] :=
A       4
B       3
C       5
D       3;
```