ISE 362 HW1

Weiming Lei

LID:882688929

5-2

DryIce should build its factories at ***Atlanta (400,000)*** and San ***Diego (200,000)***. Minimum cost is ***$129,480,000***.

AMPL Model:

```
param n>=0, integer;
param m>=0, integer;

set I =1..n ;
set J =1..m ;

var x  {j in J}        , binary ;
var y  {i in I, j in J}  >=0     ;
var z  {j in J}        , binary ;

param f {J}                    ;
param h {I}                    ;
param c {i in I, j in J}        ;
param a                        ;
param d {j in J}                ;

subject to oneplace {i in I}    : sum{j in J} y[i,j]    =  1                    ;
subject to logic    {i in I, j in J} : x[j]  >=  y[i,j]                         ;
subject to capacity {j in J}    : sum {i in I} h[i] * y[i,j] <= 200000 + a * z[j] ;

minimize cost:
sum {j in J} f[j] * x[j] + sum {i in I, j in J} c[i,j] * h[i] * y[i,j] + sum
{j in J} z[j] * d[j] ;
```

AMPL Result:

```
CPLEX 12.6.1.0: optimal integer solution; objective 129480000
36 MIP simplex iterations
0 branch-and-bound nodes
No basis.
:   x   z    :=
1   0   0
2   1   1
3   0   0
4   1   0
;

cost = 129480000
```

5-3.

a. Plant in ***United States*** produces ***125 Tons***. It supplies market in ***North America***.
   Plant in ***Germany*** produces ***475 Tons***. It supplies market in ***North America***, ***Europe***, ***Japan*** and ***Asia***.
   Plant in ***Japan*** does not produce.
   Plant in ***Brazil*** produces ***200 Tons***. It supplies market in ***North America*** and ***South America***.
   Plant in ***India*** produces ***80 Tons***. It supplies market in ***Asia***.
   (Calculation progress on next page)

b. If there is no limits on the amount produced in a plant (no upper bound and lower bound), only plants in ***Germany (420 tons)*** and ***Brazil (460 tons)*** open.
   (Calculation progress on next page)

c. ***It can***. Adding 10 tons of capacity to any of the plant which is running in full capacity will reduce costs.

d. For example, as for USD, if foreign exchange rate go up, it would result in buying from facilities in other country cheaper. For North America market, it would depend more on import so that to reduce total cost. Vice versa and the same in other currency.

5-3. a.

AMPL Model:

```
param n>=0, integer;
param m>=0, integer;

set I =1..n ;
set J =1..m ;

var x  {j in J}        , binary ;
var y  {i in I, j in J}  >=0     ;

param h {I}                    ;
param c {i in I, j in J}        ;
param p {j in J}               ;
param d {j in J}               ;

subject to demandsatisfy {i in I}: sum{j in J} y[i,j]  =  1                    ;
subject to logic {i in I, j in J} : x[j]  >=  [i,j]                            ;
subject to capacity      {j in J}        : sum {i in I} h[i] * y[i,j] <= p[j]     ;
subject to facilityran {j in J}: sum {i in I} y[i,j] * h[i] >= 0.5 * p[j] * x[j] ;

minimize cost: sum {i in I, j in J} ( c[i,j] + d[j] ) * h[i] * y[i,j]  ;
```

AMPL Result:

```
CPLEX 12.6.1.0: optimal integer solution; objective 7816450
15 MIP simplex iterations
0 branch-and-bound nodes
No basis.
x [*] :=
1  1
2  1
3  0
4  1
5  1
;

y [*,*]
:      1         2     3       4         5     :=
1   0.462963   0.5   0   0.037037   0
2   0          1     0   0          0
3   0          1     0   0          0
4   0          0     0   1          0
5   0          0.2   0   0          0.8
;

cost = 7816450
```

5-3. b.

AMPL Model:

```
param n>=0, integer;
param m>=0, integer;

set I =1..n ;
set J =1..m ;

var x  {j in J}        , binary ;
var y  {i in I, j in J}  >=0     ;

param h {I}                    ;
param c {i in I, j in J}        ;
param p {j in J}               ;
param d {j in J}               ;

subject to demandsatisfy {i in I}: sum{j in J} y[i,j]  =  1                    ;
subject to logic {i in I, j in J} : x[j]  >=  [i,j]                            ;

minimize cost: sum {i in I, j in J} ( c[i,j] + d[j] ) * h[i] * y[i,j]  ;
```

AMPL Result:

```
CPLEX 12.6.1.0: optimal integer solution; objective 7417360
0 MIP simplex iterations
0 branch-and-bound nodes
No basis.
x [*] :=
1  1
2  1
3  1
4  1
5  1
;

y [*,*]
:   1   2   3   4   5    :=
1   0   0   0   1   0
2   0   1   0   0   0
3   0   1   0   0   0
4   0   0   0   1   0
5   0   1   0   0   0
;

cost = 7417360
```

CASE STUDY (CoolWipes)

1. The annual cost of serving the entire nation from Chicago is ***$87,961,400***

2. ***I recommend adding plants.*** Open ***wipes line*** in ***Princeton*** and ***Los Angeles.*** Cost would down to ***$86,325,400***
   When the costs are half of their current value, ***I do not recommend adding plants***.
   When the costs are double of their current value, ***I recommend adding plants.*** Open ***wipes line*** in ***Princeton*** and ***Los Angeles.***

AMPL Model:

```
param n>=0, integer;
param m>=0, integer;

set I =1..n ;
set J =1..m ;

var x {j in J}, binary;
var y {i in I, j in J} >=0;

param f  {j in J};
param co {i in I, j in J};
param p  {j in J};
param h  {i in I};
param ca {j in J};

subject to demandsayisfy {i in I}: sum {j in J} y[i,j] = 1;
subject to facilityopen  {i in I, j in J}: x[j]>=y[i,j];
subject to capacity {j in J}: sum { i in I} h[i]* y[i,j] <= ca[j];
subject to chicago : x[1]=1;


  minimize cost: sum {j in  J} f[j] * x[j] + sum{i in I, j in J} ( co[i,j] +
p[j] ) * h[i] * y[i,j];
```

AMPL Result:

Original Cost

Wipes line

```
x [*] :=
1  1
2  1
3  0
4  1
;

y [*,*]
:   1   2   3   4    :=
1   0   0   0   1
2   0   0   0   1
3   1   0   0   0
4   1   0   0   0
5   0   1   0   0
6   0   1   0   0
;

cost = 71828000
```

ointment line

```
x [*] :=
1  1
2  0
3  0
4  0
;

y [*,*]
:   1   2   3   4    :=
1   1   0   0   0
2   1   0   0   0
3   1   0   0   0
4   1   0   0   0
5   1   0   0   0
6   1   0   0   0
;

cost = 14497400
```

Half Cost

Wipes line

```
x [*] :=
1  1
2  0
3  0
4  0
;

cost = 61732000
```

ointment line

```
x [*] :=
1  1
2  0
3  0
4  0
;

cost = 13148700
```

Double Cost

Wipes line

```
x [*] :=
1  1
2  1
3  0
4  1
;

y [*,*]
:   1   2   3   4    :=
1   0   0   0   1
2   0   0   0   1
3   1   0   0   0
4   1   0   0   0
5   0   1   0   0
6   0   1   0   0
;

cost = 89256000
```

ointment line

```
x [*] :=
1  1
2  0
3  0
4  0
;

y [*,*]
:   1   2   3   4    :=
1   1   0   0   0
2   1   0   0   0
3   1   0   0   0
4   1   0   0   0
5   1   0   0   0
6   1   0   0   0
;

cost = 17194800
```

CASE STUDY (CoolWipes)

3. I would recommend open ***wipes line*** in ***Princeton, Atlanta*** and ***Los Angeles.*** Cost would be ***$70,668,000.*** And open ***ointment line*** in ***Atlanta.*** Cost would be ***$14,434,400.***
   When the costs are half of their current value, I would recommend open ***wipes line*** in ***Princeton, Atlanta*** and ***Los Angeles.*** Cost would be ***$61,134,000.*** And open ***ointment line*** in ***Atlanta.*** Cost would be ***$13,117,200.***
   When the costs are double of their current value, I would recommend open ***wipes line*** in ***Chicago, Princeton*** and ***Los Angeles.*** Cost would be ***$89,256,000.*** And open ***ointment line*** in ***Atlanta.*** Cost would be ***$17,068,800.***

AMPL Model:

```
param n>=0, integer;
param m>=0, integer;

set I =1..n ;
set J =1..m ;

var x {j in J}, binary;
var y {i in I, j in J} >=0;

param f  {j in J};
param co {i in I, j in J};
param p  {j in J};
param h  {i in I};
param ca {j in J};

subject to demandsayisfy {i in I}: sum {j in J} y[i,j] = 1;
subject to facilityopen  {i in I, j in J}: x[j]>=y[i,j];
subject to capacity {j in J}: sum { i in I} h[i]* y[i,j] <= ca[j];

minimize cost: sum {j in  J} f[j] * x[j] + sum{i in I, j in J} ( co[i,j] + p[j] )
* h[i] * y[i,j];
```

AMPL Result:

Original Cost

Wipes line

```
x [*] :=
1  0
2  1
3  1
4  1
;

y [*,*]
:   1   2   3   4    :=
1   0   0   0   1
2   0   0   0   1
3   0   1   0   0
4   0   0   1   0
5   0   1   0   0
6   0   0   1   0
;

cost = 70668000
```

ointment line

```
x [*] :=
1  0
2  0
3  1
4  0
;

y [*,*]
:   1   2   3   4    :=
1   0   0   1   0
2   0   0   1   0
3   0   0   1   0
4   0   0   1   0
5   0   0   1   0
6   0   0   1   0
;

cost = 14434400
```

Half Cost

Wipes line

```
x [*] :=
1  0
2  1
3  1
4  1
;

y [*,*]
:   1   2   3   4    :=
1   0   0   0   1
2   0   0   0   1
3   0   1   0   0
4   0   0   1   0
5   0   1   0   0
6   0   0   1   0
;

cost = 61134000
```

ointment line

```
x [*] :=
1  0
2  0
3  1
4  0
;

y [*,*]
:   1   2   3   4    :=
1   0   0   1   0
2   0   0   1   0
3   0   0   1   0
4   0   0   1   0
5   0   0   1   0
6   0   0   1   0
;

cost = 13117200
```

Double Cost

Wipes line

```
x [*] :=
1  1
2  1
3  0
4  1
;

y [*,*]
:   1   2   3   4    :=
1   0   0   0   1
2   0   0   0   1
3   1   0   0   0
4   1   0   0   0
5   0   1   0   0
6   0   1   0   0
;

cost = 89256000
```

ointment line

```
x [*] :=
1  0
2  0
3  1
4  0
;

y [*,*]
:   1   2   3   4    :=
1   0   0   1   0
2   0   0   1   0
3   0   0   1   0
4   0   0   1   0
5   0   0   1   0
6   0   0   1   0
;

cost = 17068800
```