EVALUATION GUIDELINES - Written examination

GRA 62274
Business Optimisation

Department of Accounting, Auditing and Business Analytics

Start date:	19.12.2018	Time 09:00
Finish date:	19.12.2018	Time 12:00

For more information about formalities, see examination paper.

Problem 1 (35 %)

A product is produced at three factories (A, B, and C) and shipped to four markets (1, 2, 3, and 4).

Capacities at the three factories:

A	***B***	***C***
100	140	110

Demand at the four markets:

1	***2***	***3***	***4***
55	60	35	50

Variable cost per unit for production and shipping:

	To ***1***	To ***2***	To ***3***	To ***4***
From ***A***	7	4	9	6
From ***B***	3	8	5	4
From ***C***	6	5	4	11

(In the following, try to avoid non-linear formulations if possible.)

a) Use the given data and develop a linear programming model (either a mathematical model or an AMPL model) for minimizing total costs. All demand must be satisfied.

AMPL model:

```
set FACT;
set CUST;
param cap {i in FACT};
param d {j in CUST};
param c {i in FACT, j in CUST};
var X {i in FACT, j in CUST} >= 0;

minimize TotalCosts:      sum {i in FACT, j in CUST} c[i,j] * X[i,j];
s.t. C1 {i in FACT}:      sum {j in CUST} X[i,j] <= cap[i];
s.t. C2 {j in CUST}:      sum {i in FACT} X[i,j] = d[j];

data;
set FACT := A B C;
set CUST := 1 2 3 4;
param cap :=
  A 100  B 140  C 110 ;
param d :=
  1  55
  2  60
  3  35
  4  50 ;
param c:
        1     2     3     4     :=
 A      7     4     9     6
 B      3     8     5     4
 C      6     5     4     11    ;
```

b) The company wants to produce this product at fewer factories. For each factory, there is a fixed cost associated with producing the product at the factory, as follows.

Fixed cost per factory:

A	*B*	*C*
80	90	100

The fixed cost is avoided if the product is not produced at the factory.
Modify your model from a), so that the sum of variable and fixed costs is minimized.

New parameter for fixed costs:
```
param f {i in FACT};
```

New variable:
```
var Y {i in FACT} binary;
```

Modified objective function:
```
minimize TotalCosts:
            sum {i in FACT} f[i] * Y[i]
      +     sum {i in FACT, j in CUST} c[i,j] * X[i,j];
```

Modified capacity constraints:
```
s.t. C1 {i in FACT}:     sum {j in CUST} X[i,j] <= cap[i]*Y[i];
```

New data:
```
param f :=  A 80  B 90  C 100 ;
```

c) The company wants to test out an alternative approach, in which fixed costs are not included in the model. Instead, the company wants a solution in which each factory will produce either zero of the product, or a quantity that is at or above a minimum level.

The minimum level for each factory is as follows:

A	*B*	*C*
70	80	75

(That is, for factory A, the total quantity must be at least 70 units, or otherwise it must be zero.)

Modify your model from a) to create an alternative model that fits the above description.

New parameter for minimum requirements at each factory:
```
param minreq {i in FACT};
```

New variable (as in b)):
```
var Y {i in FACT} binary;
```

Capacity constraints (as in b)):
```
s.t. C1 {i in FACT}:     sum {j in CUST} X[i,j] <= cap[i]*Y[i];
```

Constraints to ensure minimum quantity if the factory is used:

```
s.t. C3 {i in FACT}:      sum {j in CUST} X[i,j] >= minreq[i]*Y[i];
```

New data (minimum requirements):

```
param minreq :=  A 70  B 80  C 75 ;
```

d) After having studied the solution from c), the company wants to impose a minimum quantity of 20 units on the shipments between each factory and market. That is, the quantity must be at or above 20, or it must be zero.
For example, the quantity shipped from factory B to market 3 must be at least 20, or it must be 0.

Modify your model from c) so that it incorporates the new restrictions.

New parameter for minimum requirements between factory and market:

```
param minreq2 {i in FACT, j in CUST};
```

New variable that indicates shipments from a factory to a market:

```
var Z {i in FACT, j in CUST} binary;
```

Constraints to ensure that the binary variable is equal to 1 if there are shipments from a factory to a market:

```
s.t. C4 {i in FACT, j in CUST}: X[i,j] <= cap[i]*Z[i,j];
```

Constraints to ensure minimum quantity between factory and market:

```
s.t. C5 {i in FACT, j in CUST}: X[i,j] >= minreq2[i,j]*Z[i,j];
```

New data (minimum requirements between factory and market):

```
param minreq2:
       1     2     3     4     :=
 A     20    20    20    20
 B     20    20    20    20
 C     20    20    20    20    ;
```

Problem 2 (30 %)

A company is attempting to choose among a series of investment alternatives. The potential investment alternatives, the net present value of the future stream of returns, the capital requirements, and the available capital funds over the next 3 years are summarized as follows:

Alternative	Net present value	Capital requirements		
		Year 1	*Year 2*	*Year 3*
Project 1	67	24	26	22
Project 2	33	40	10	32
Project 3	18	32	14	17
Project 4	41	13	15	31
Project 5	26	30	19	28
Capital funds available:		85	80	75

The investments cannot be postponed. Hence, the timing of the investments is not an issue here.

Partial investment (investing in less than 100% of a project) is not possible.

(In the following, try to avoid non-linear formulations if possible.)

a) Develop an AMPL model for maximizing the total net present value.
(You are not supposed to find the optimal solution; just formulate the model.)

```
set YEAR;
set ASSET;

param funds {YEAR};
param npv {ASSET};
param capreq {ASSET, YEAR};

var X {ASSET} binary;

maximize TotalValue: sum {a in ASSET} npv[a]*X[a];
s.t. C1 {y in YEAR}: sum {a in ASSET} capreq[a,y]*X[a] <= funds[y];

data;
set YEAR := 1 2 3;
set ASSET := 1 2 3 4 5;
param npv :=
      1      67
      2      33
      3      18
      4      41
      5      26 ;
param funds := 1 85 2 80 3 75;
```

```
param capreq:
        1       2       3       :=
1       24      26      22
2       40      10      32
3       32      14      17
4       13      15      31
5       30      19      28 ;
```

b) Modify your model from a) to incorporate the following:

Of projects 1, 3, and 4, at least two must be chosen.

```
s.t. C2:     X[1] + X[3] + X[4] >= 2;
```

c) Modify your model from a) to incorporate the following:

If project 2 is chosen, project 4 cannot be chosen.

```
s.t. C2:     X[2] + X[4] <= 1;
```

d) Modify your model from a) to incorporate the following:

If both projects 4 and 5 are chosen, project 3 must also be chosen.

```
s.t. C2:     X[3] >= X[4] + X[5] - 1;
```

e) Modify your model from a) to incorporate the following:

If both project 2 and 5 are chosen, exactly one of project 1 and 4 must be chosen.

```
s.t. C2:     X[1] + X[4] >= X[2] + X[5] - 1;
s.t. C3:     X[1] + X[4] <= 3 - X[2] - X[5];
```

Problem 3 (35 %)

At many educational institutions, scheduling of examinations is a challenging task. The problem consists of assigning an examination date and timeslot for each subject, such that specific criteria and constraints are taken into account.

During the last decades, there has been a significant amount of research in which these problems are formulated as optimization models.

Problem description:

A number of candidates (students) have signed up for a number of specified subjects (mathematics, decision theory, microeconomics, etc.).

There are a number of available dates, and on each date there are multiple timeslots (morning, afternoon, etc.), so that there can be examinations in multiple subjects on the same date.

Below is an example of an optimization model.

C is the number of candidates, *S* is the number of subjects, *D* is the number of available dates, and *T* is the number of available timeslots per date. In this model, there can be only one subject per timeslot.

The variables of this model are:

$X_{c,s,d,t}$	A binary variable which is 1 if candidate *c* has an examination in subject *s*, on date *d*, timeslot *t.* Otherwise zero.
$Y_{s,d,t}$	A binary variable which is 1 if there is an examination of subject *s*, on date *d*, timeslot *t.* Otherwise zero.

The parameters are:

f_d	A cost per examination. The cost depends on the date, e.g., Saturdays are more expensive than other weekdays.
$a_{c,s}$	A parameter that is 1 if candidate c is enrolled in (has signed up for) subject *s*.

The model:

Minimize
$$\sum_{s=1}^{S}\sum_{d=1}^{D}\sum_{t=1}^{T} f_d Y_{s,d,t} \qquad (1)$$

subject to:
$$Y_{s,d,t} \geq X_{c,s,d,t} \qquad \text{for all } c, s, d \text{ and } t \qquad (2)$$

$$\sum_{d=1}^{D}\sum_{t=1}^{T} X_{c,s,d,t} = a_{c,s} \qquad \text{for all } c \text{ and } s \qquad (3)$$

$$\sum_{d=1}^{D}\sum_{t=1}^{T} Y_{s,d,t} = 1 \qquad \text{for all } s \qquad (4)$$

$$\sum_{s=1}^{S} Y_{s,d,t} \leq 1 \qquad \text{for all } d \text{ and } t \qquad (5)$$

a) Explain briefly the objective function and the constraints of the model.

(1) Minimizing total costs. The fixed cost f_d is incurred every time $Y_{s,d,t} = 1$.
(2) The binary variable $Y_{s,d,t}$ is forced to be 1 if there is a candidate having an examination in subject *s*, on day *d* and in timeslot *t*.
(3) For candidate *c* enrolled for subject *s*, there is exactly one timeslot in which the examination takes place.
(4) For each subject, there is exactly one timeslot in which the examination takes place.
(5) For each timeslot, there is at most one examination.

(In the following, try to avoid non-linear formulations if possible.)

b) Every year, it happens that some students need to take two examinations on the same date (in two different timeslots). Add a constraint that ensures that such "doubles" will never happen.

$$\sum_{s=1}^{S}\sum_{t=1}^{T} X_{c,s,d,t} \leq 1 \qquad \text{for all } c \text{ and } d$$

c) The new constraint from b) may make your model infeasible, that is, there may exist no solution in which such "doubles" can be avoided. Instead of using the constraint from b), suggest a new objective function that will minimize the number of "doubles". Do not take costs into consideration in this case.

Let $Z_{c,d}$ be a binary variable which is 1 if there are multiple examinations for candidate *c* on day *d*.

$$\text{Minimize} \qquad \sum_{c=1}^{C}\sum_{d=1}^{D} Z_{c,d}$$

$$Z_{c,d} \geq \frac{1}{T}\left(\sum_{s=1}^{S}\sum_{t=1}^{T} X_{c,s,d,t} - 1\right) \qquad \text{for all } c \text{ and } d \qquad (*)$$

Constraint (*) forces $Z_{c,d}$ to be 1 when there is more than one examination for candidate *c* on day *t*.

d) Now, disregard the problem of “doubles”. That is, we go back to the assumptions in a). The school administration is concerned about the total number of dates being used for examinations. Modify the model, so that the total number of dates in which there is an examination, is minimized. Do not include costs in this model.

Let W_d be a binary variable which is 1 if any examination takes place on day d.

Minimize $$\sum_{d=1}^{D} W_d$$

$$W_d \geq Y_{s,d,t} \qquad \text{for all } s, d \text{ and } t$$

(In the appendix, a corresponding AMPL model is shown. You can choose whether you want to base your answers on the AMPL model - and modify the AMPL model - instead of the above mathematical model.)

Appendix

AMPL model for Problem 3:

```
param C;
param S;
param D;
param T;
param f{1..D};
param a{1..C, 1..S};

var X{1..C, 1..S, 1..D, 1..T} binary;
var Y{1..S, 1..D, 1..T} binary;

minimize totalCosts:
      sum{s in 1..S, d in 1..D, t in 1..T} f[d]*Y[s,d,t];

s.t. c2 {c in 1..C, s in 1..S, d in 1..D, t in 1..T}:
      Y[s,d,t] >= X[c,s,d,t];

s.t. c3 {c in 1..C, s in 1..S}:
      sum {d in 1..D, t in 1..T} X[c,s,d,t] = a[c,s];

s.t. c4 {s in 1..S}:
      sum {d in 1..D, t in 1..T} Y[s,d,t] = 1 ;

s.t. c5 {d in 1..D, t in 1..T}:
      sum {s in 1..S} Y[s,d,t] <= 1 ;
```