Homework 8 Solutions

In preparation for Exam 2, you should grade yourself using these solutions and find what areas have the greatest needs for improvement, and do more practice in those areas.

The solutions here give you an indication of how much to write in the exam for each type of question. The solutions introduce various shorthands, both in writing formulations and in the coding, which can save you time.

Exam 2 Sample A (100 Points)

Q1. Multiple Choice (10 points)

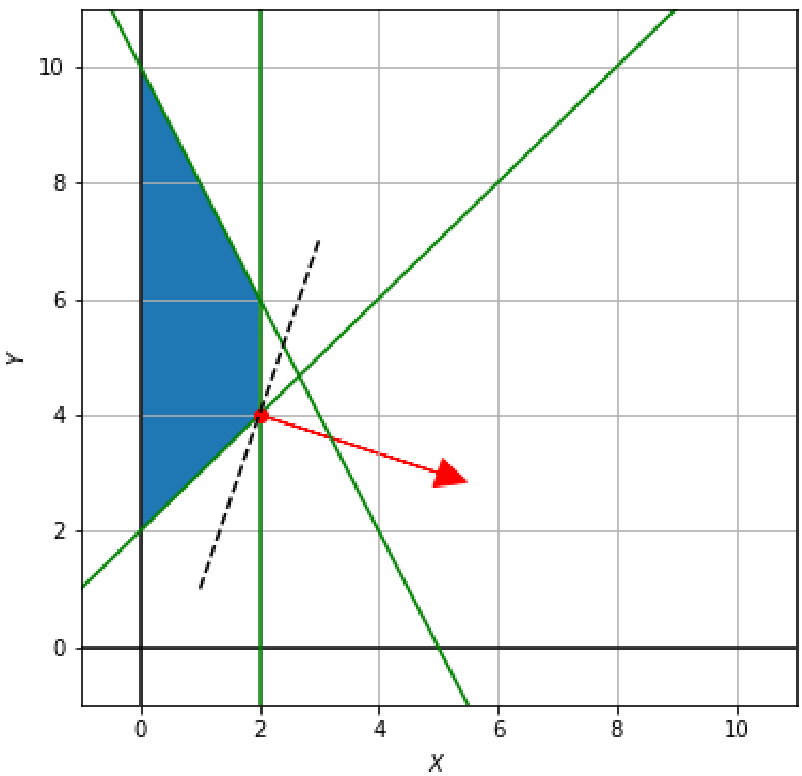
i) Consider the following linear program, which has an unique optimal solution.

$$\begin{aligned} \text{Maximize:} \quad & 3X - Y \\ \text{subject to:} \quad & \\ & 2X + Y \leq 10 \\ & X + 2 \leq Y \\ & X \leq 2 \end{aligned}$$

What is the optimal value of decision variable Y?

A) 2
ANSWER B) 4
C) 6
D) 10
E) None of the above

Solution: After graphing out the feasible region and the direction of the objective function, we find that the optimal solution is the intersection of the lines $X + 2 = Y$ and $X = 2$. Solving, we get $Y = 4$. See diagram below.



ii) Suppose that X and Y are positive **input parameters (data with known numerical values)** and Z and W are **decision variables**, which of the following **CANNOT be the objective function** of a linear program (LP)?
A) $\frac{X}{Y}Z + \frac{Y}{X}W$
B) $(X - Y)(Z + X)$
C) $W + XZ$
ANSWER D) WZ
E) XY
Solution: An objective function of a linear program must be **linear in the decision variables.** We highlight the decision variables below in bold, to help us identify any violation.
A) $\frac{X}{Y}\mathbf{Z} + \frac{Y}{X}\mathbf{W}$
B) $(X - Y)(\mathbf{Z} + X)$
C) $\mathbf{W} + X\mathbf{Z}$
D) $\mathbf{W}Z$
E) XY
As can be seen, option D) has two decision variables multiplied with one another, which makes it not a linear function of decision variables.

Q2. Concrete Modeling (30 points)

A food factory requires 2000 tons of canola oil every month as a raw ingredient. The price of canola oil fluctuates from month to month due to market conditions. The predicted prices for the next six months are as follows

	Apr	May	Jun	Jul	Aug	Sep
Price ($) per ton	150	160	180	170	180	160

The factory's supplier for canola oil delivers it on the first day of every month, and charges the prices above. The factory can decide how much oil to buy each month from the supplier. The factory can also store up to 1000 tons of oil for future use. (The current inventory before the shipment in April is zero.) Formulate an optimization problem to decide how much canola oil to buy for each of the six months in order to minimize the total purchase cost over these six months.
a) What is the decision (in words)? (2 points)
How much canola oil to buy for each of the six months.
b) What is the objective (in words)? (2 points)
Minimize the total purchase cost.
c) Describe any three constraints (in words). (6 points)
Any three of the following would do:

- At most 1000 tons saved.
- 2000 tons of oil needed every month.
- Non-negative inventory of oil.
- Inventory balance from month to month.
- Current inventory of zero.
- Non-negative quantity of oil bought each month.

d) Write a concrete LP formulation using the correct mathematical notation, specifying the decision variables, the objective function and all constraints. (20 points)
Decision Variables:

- $X_4, X_5, \cdots, X_9$: amount of oil to buy in each month. (continuous)

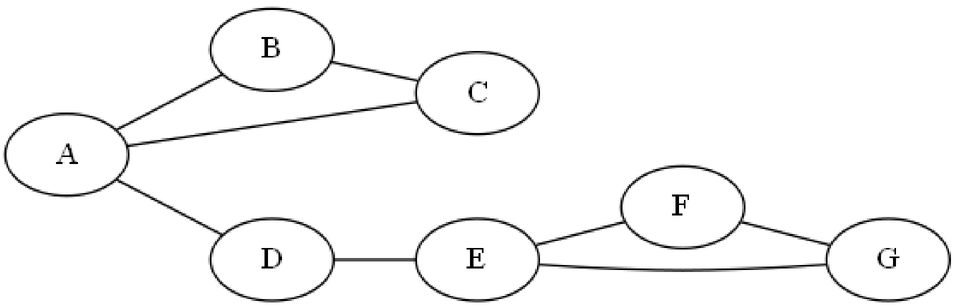
- $Y_4, Y_5, \cdots, Y_9$: amount of oil stored at the end of each month. (continuous)

$$
\begin{aligned}
\text{Min} \quad & 150X_4 + 160X_5 + 180X_6 + 170X_7 + 180X_8 + 160X_9 \\
\text{s. t.} \quad & \\
& Y_4 = X_4 - 2000 \\
& Y_5 = Y_4 + X_5 - 2000 \\
& Y_6 = Y_5 + X_6 - 2000 \\
& Y_7 = Y_6 + X_7 - 2000 \\
& Y_8 = Y_7 + X_8 - 2000 \\
& Y_9 = Y_8 + X_9 - 2000 \\
& Y_t \leq 1000 \qquad \text{for each month } t. \\
& X_t, Y_t \geq 0
\end{aligned}
$$

In the exam, for non-negativity constraints, you can omit qualifications such as "for each month t" when it is obvious.

Q3. Abstract Modeling (35 points)

Ebony is an ambitious master's student who would like to maximize the number of extra-curricular business analytics projects she takes part of this year. However, projects may conflict with one another. The following graph summarizes the conflicts. (For example, project A conflicts with B, C and D, but projects B and D can be done together.)



Beside the conflict above,

- Project A is a prerequisite to project F (meaning that pursuing F requires also pursuing A.)
- Project B is a prerequisite to project G.

Formulate an optimization problem to help her decide which projects to pursue.
a) What is the decision (in words)? (2 points)
Which projects to pursue.
b) What is the objective (in words)? (2 points)
Maximize # of projects pursued.
c) Except for the prerequisite constraints, describe any one constraint (in words). (2 points)
Projects A and B cannot be done together.
d) Write a concrete MIP formulation using the correct mathematical notation, specifying the decision variables, the objective function and all constraints. (15 points)
DV: X_i: whether to pursue project i. (binary)

$$
\begin{aligned}
\text{Maximize} \quad & X_A + X_B + \cdots + X_G \\
\text{s.t.} \quad & \\
& X_A + X_B \leq 1 \\
& X_B + X_C \leq 1 \\
& X_A + X_C \leq 1 \\
& X_A + X_D \leq 1 \\
& X_D + X_E \leq 1 \\
& X_E + X_F \leq 1 \\
& X_F + X_G \leq 1 \\
& X_E + X_G \leq 1 \\
& X_F \leq X_A \\
& X_G \leq X_B
\end{aligned}
$$

Note that we specified the decision variables as binary when defining them. If we didn't do that, then we also need the constraint

$$X_A, X_B, \cdots, X_G \in \{0, 1\}$$

Another possible formulation is to replace the first three inequalities and the inequalities by

$$X_A + X_B + X_C \leq 1$$

and similarly the 3 inequalities on mutual exclusivity of X_E, X_F, and X_G by $X_E + X_F + X_G \leq 1$.

e) Suppose that I is the set of projects. Define data variables to encode all inputs needed when generalizing the above MIP formulation to arbitrarily large data sets. (4 points)

Define a_{ij} for whether projects i and j can be done together. (binary)

f) Write an abstract MIP formulation using the data variables you defined in part e), specifying the decision variables, the objective function and all constraints. (10 points)

DV: X_i is whether to pursue project i. (binary)

$$
\begin{aligned}
\text{Maximize} \quad & \sum_{i \in I} X_i \\
\text{s.t.} \quad & \\
& X_i + X_j \leq 1 \qquad \text{for all } (i, j) \text{ such that } a_{ij} = 1. \\
& X_F \leq X_A \\
& X_G \leq X_B
\end{aligned}
$$

Solution 2 for f):

The question was unclear about whether you were to generalize the prerequisite constraints or not. The above answer does not generalize them. The following formulation generalizes them as well.

Data:

- a_{ij}: whether projects i and j can be done together. (binary)
- b_{ij}: whether project i is a prerequisite of project j. (binary)

DV: X_i is whether to pursue project i. (binary)

$$
\begin{aligned}
\text{Maximize} \quad & \sum_{i \in I} X_i \\
\text{s.t.} \quad & \\
& X_i + X_j \leq 1 \qquad \text{for all } (i, j) \text{ such that } a_{ij} = 1. \\
& X_j \leq X_i \qquad \text{for all } (i, j) \text{ such that } b_{ij} = 1.
\end{aligned}
$$

Q4. Gurobi Coding (25 points)

The following MIP is used to assign students into project teams in this course, to balance the overall characteristics of each team.

Data:

- I: set of students.
- n: number of teams
- $J = \{1, 2, \cdots, n\}$: set of teams.
- K: set of characteristics.
- a_{ik}: student i's value for characteristics k.
- w_k: the weight for characteristics k in the objective.
- L_k: the best lower bound for the sum of characteristic k for any team.
- U_k: the best upper bound for the sum of characteristics k for any team.

For example, suppose that $n = 2$, and the data is stored in an excel file called `data.xlsx` with two sheets. The sheet named `Sheet 1` encodes I, K and a_{ik}'s. In the below, $I = \{A, B, C, D, E, F\}$, and $K = \{Person, Male, Programmer, Math, Speaking\}$.

	A	B	C	D	E	F
1	Names	Person	Male	Programmer	Math	Speaking
2	A	1	1	1	1	1
3	B	1	1	0	0	0
4	C	1	0	0	1	1
5	D	1	0	0	1	0
6	E	1	0	0	0	1
7	F	1	1	1	0	1
8						

The sheet named `Sheet 2` encodes the w_k, L_k and U_k for each characteristic k.

	A	B	C	D	E	F
1		Person	Male	Programmer	Math	Speaking
2	Weights	10	3	5	2	2
3	L	3	1	1	1	2
4	U	3	2	1	2	2

Decision variables:

- x_{ij} : whether to assign student i to team j. (Binary)
- s_k, and t_k : auxilliary variables for each characteristic k. (Continuous)

MIP:

$$\text{Minimize:} \quad \sum_{k \in K} w_k (s_k + t_k)$$

subject to:

$$\text{(Every person assigned)} \quad \sum_{j \in J} x_{ij} = 1 \quad \text{For each person } i \in I.$$

$$\text{(Team balance)} \quad L_k - s_k \leq \sum_{i \in I} a_{ik} x_{ij} \leq U_k + t_k \quad \text{For each team } j \in J \text{ and each } k \in K.$$

$$\text{(Binary)} \quad x_{ij} \in \{0, 1\} \quad \text{for all } i \text{ and } j.$$

$$\text{(Non-negativity)} \quad s_k, t_k \geq 0 \quad \text{for all } k.$$

Complete the following code to implement the above MIP. You should store the answer in a DataFrame named `answer` with one column called `Team`: the row index is the name of each person and the column `Team` stores the team number j to which the person is assigned.

Helpful shorthands:

The following solution illustrate the use of shorthand `mod.addVars`, which allows you to save the amount of material to write on the exam. Instead of

```
x={}
for i in I:
    for j in J:
        x[i,j]=mod.addVar(vtype=grb.GRB.BINARY)
```

You can do the following one-liner, and the result is equivalent. Note the "s" at the end of `addVars`, which is what distinguishes it from the above.

```
x=mod.addVars(I,J,vtype=grb.GRB.BINARY)
```

Similarly, instead of

```
s={}
for k in K:
    s[k]=mod.addVar(lb=0)
```

You can do the equivalent

```
s=mod.addVars(K,lb=0)
```

The syntax is to first supply the indices (as in `I, J` in the first example, and `K` in the second example), then the argument you want. You can also supply a `name` parameter, and the variables will be named something like `XXX[0,1]`, where `XXX` is the `name` you specify and inside the square brackets are the values of indices corresponding to the variable.

Another shorthand is that the default sense of the Gurobi `setObjective` function is to minimize, so you can omit the `sense=grb.GRB.MINIMIZE`.

```
[9]: import pandas as pd
     sheet1=pd.read_excel('Q4.xlsx',sheet_name='Sheet 1',index_col=0)
     sheet2=pd.read_excel('Q4.xlsx',sheet_name='Sheet 2',index_col=0)
     n=2

     # Write your code below
     import gurobipy as grb
     mod=grb.Model()

     # Preparing data
     I=sheet1.index
     J=range(1,n+1)
     K=sheet1.columns

     # Defining decision variables
     x=mod.addVars(I,J, vtype=grb.GRB.BINARY)
     s=mod.addVars(K,lb=0)
```

```python
t=mod.addVars(K,lb=0)

# Defining objective and constraints, following the formulation line by line
mod.setObjective(sum(sheet2.loc['Weights',k]*(s[k]+t[k]) for k in K))
for i in I:
    mod.addConstr(sum(x[i,j] for j in J)==1)
for j in J:
    for k in K:
        mod.addConstr(sum(sheet1.loc[i,k]*x[i,j] for i in I)>=sheet2.loc['L',k]-s[k])
        mod.addConstr(sum(sheet1.loc[i,k]*x[i,j] for i in I)<=sheet2.loc['U',k]+t[k])
mod.optimize()

data=[]
for i in I:
    for j in J:
        if x[i,j].x>0:
            data.append([j])
            break
answer=pd.DataFrame(data,columns=['Team'],index=I)
answer
```

```
      Team
Names
A        2
B        2
C        2
D        1
E        1
F        1
```

Exam 2 Sample B (100 Points)

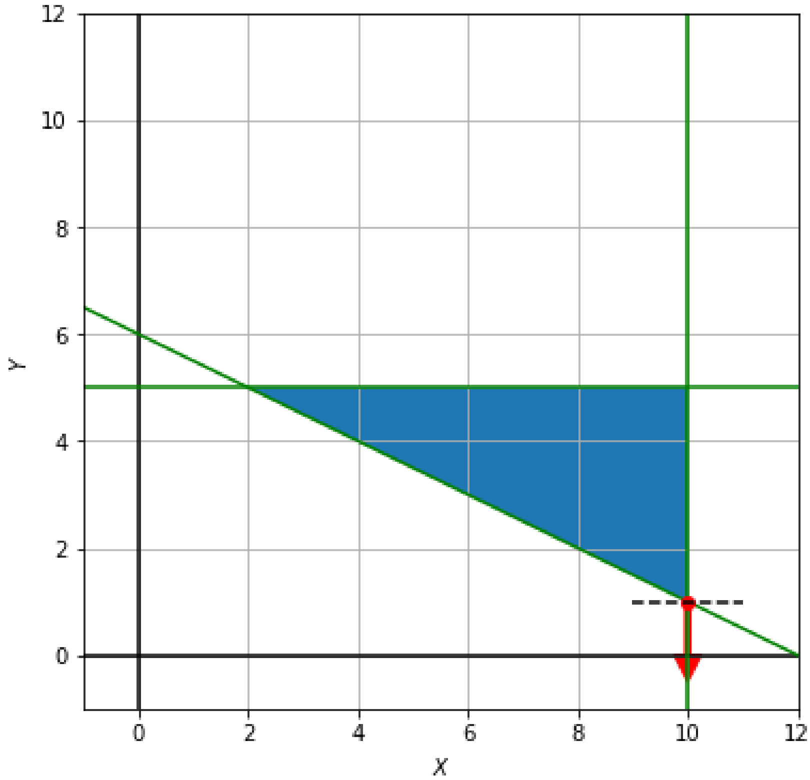
Q5. Multiple Choice (10 points)

i) Consider the following linear program.

$$
\begin{array}{rr}
\text{Minimize:} & Y \\
\text{subject to:} & \\
\text{(C1)} & X + 2Y \geq 12 \\
\text{(C2)} & X \leq 10 \\
\text{(C3)} & Y \leq 5 \\
\text{(C4)} & X \geq 0 \\
\text{(C5)} & Y \geq 0
\end{array}
$$

What are the binding constraints?
ANSWER A) C1 and C2.
B) C2 and C3.
C) C1 and C3.
D) C3 and C4.
E) C2 and C5.

Solution: After drawing out the feasible region and the direction of the optimal objective, we see that the optimal solution is the intersection of the lines $X = 10$ (C2) and $X + 2Y = 12$ (C1). See diagram below.



ii) Suppose there are two factories A and B, and two demand centers 1 and 2. Goods are produced in the two factories and shipped to the demand centers. The following LP minimizes the total transportation cost (measured in thoursands of dollars) subject to respecting the capacity of each factory and fulfilling all demand.

$$\begin{aligned}
\text{Minimize:} \quad & 5X_{A1} + 3X_{A2} + 4X_{B1} + 7X_{B2} \\
\text{subject to:} \quad & \\
\text{(Capacity A)} \quad & X_{A1} + X_{A2} \leq 10 \\
\text{(Capacity B)} \quad & X_{B1} + X_{B2} \leq 8 \\
\text{(Demand 1)} \quad & X_{A1} + X_{B1} \geq 9 \\
\text{(Demand 2)} \quad & X_{A2} + X_{B2} \geq 8 \\
& X_{A1}, X_{A2}, X_{B1}, X_{B2} \geq 0
\end{aligned}$$

The shadow price of the (Capacity B) constraint is -1, which is valid when the RHS is between 7 and 9. The shadow price of the (Demand 1) constraint is 5, which is valid when the RHS is between 8 and 10. **Which of the following statements is TRUE?**

A) Increasing the capacity of factory B by 2 units would decrease the transportation cost by 2 units.

B) Increasing the demand requirement at center 1 would reduce the transportation cost.

ANSWER C) Increasing the demand requirement of center 1 by one unit would change the optimal value of decision variables X_{A1} or X_{B1}.

D) If we simultaneously reduce the demand requirement at center 1 to 8 and increase the capacity of factory B to 9, then the total transportation cost would decrease by 6 units.

E) If it is possible to pay another company 6 thousand dollars in order to satisfy one unit of demand at center 1, then we should pay them to do it.

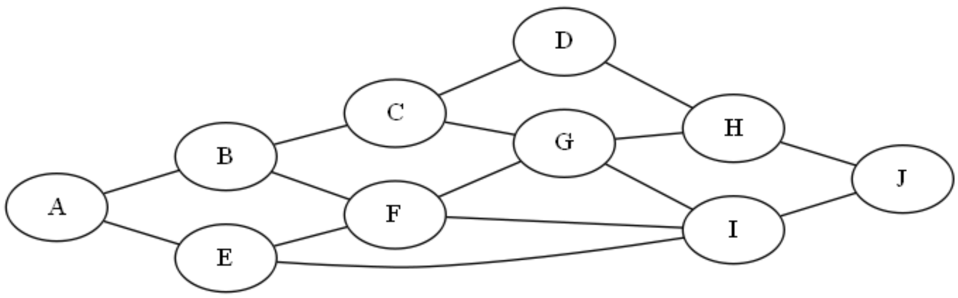
Solution: Option A is incorrect because increasing by 2 unit would be outside the allowable range. Option B is incorrect because increasing the demand requirement would increase the cost by 5, not reduce it.

Option C is correct because changing the demand rquirement of center 1 by one unit is within the allowable range, so the constraint remains binding. Hence $X_{A1} + X_{B1} = 10$, which is different from the current $X_{A1} + X_{B1} = 9$. Hence, one of X_{A1} or X_{B1} must change.

Option D is incorrect because shadow prices are only valid when we change one constraint at a time, holding the other fixed. Option E is incorrect because reducing the demand requirement at center 1 by one unit would save us only 5 thousand dollars in terms of cost, so the savings do not justify paying 6 thousand dollars for the demand reduction.

Q6. Concrete Modeling (30 points)

USC would like to protect every intersection around USC by stationing security staff, so that every intersection either has a staff stationed, or neighbors another intersection that has a staff stationed. For example, in the following sample map with 10 intersections, the neighbors of intersection A are B and E. So if a staff is stationed at intersection A, then intersections A, B, and E are all protected.



For the above map, formulate an optimization problem (LP or MIP) to minimize the number of staff needed, subject to protecting all 10 intersections, as well as satisfying the following constraints:

- Staff cannot be stationed at both intersections A and B.
- At least three of E, F, G and H must have a staff directly stationed.
- No one can be stationed at intersection J.
- There must be at least one staff who is stationed at an intersection where he/she can simultaneously protect both intersections C and H.

a) What is the decision (in words)? (3 points)

Which intersections to station staff.

b) What is the objective (in words)? (2 points)

Minimize the # of staff stationed.

c) Other than the constraints listed in bullets above, describe any one constraint (in words). (3 points)

At least 1 staff stationed among A, B and E (in order to protect intersection A).

d) Write a concrete LP formulation using the correct mathematical notation, specifying the decision variables, the objective function and all constraints. (22 points)

DV: $X_A, X_B, \cdots, X_J$ for whether to station staff at each intersection. (binary)

$$
\begin{aligned}
\text{Min.} \quad & X_A + X_B + \cdots + X_J \\
\text{s.t.} \quad & X_A + X_B + X_E \geq 1 \\
& X_A + X_B + X_C + X_F \geq 1 \\
& X_B + X_C + X_D + X_G \geq 1 \\
& X_C + X_D + X_H \geq 1 \\
& X_A + X_E + X_F + X_I \geq 1 \\
& X_B + X_E + X_F + X_G + X_I \geq 1 \\
& X_C + X_F + X_G + X_H + X_I \geq 1 \\
& X_D + X_G + X_H + X_J \geq 1 \\
& X_E + X_F + X_G + X_I + X_J \geq 1 \\
& X_H + X_I + X_J \geq 1 \\
& X_A + X_B \leq 1 \\
& X_E + X_F + X_G + X_H \geq 3 \\
& X_J = 0 \\
& X_D + X_G \geq 1
\end{aligned}
$$

Q7. Abstract Modeling (35 points)

Luke would like to optimize his diet to minimize cost while satisfying minimal nutritional requirements. The first table below shows his favorite foods, as well as the nutrition information per serving and cost per servint. The second table shows the requirement on his total daily in-take of each nutrient.

Foods	Calories	Protein	Fat	Sodium	Costs
1. ice cream	330	8	15	180	$1.59
2. chicken	420	32	10	300	$2.89
3. pizza	320	15	20	820	$1.99
4. fries	380	4	19	270	$1.89
5. macaroni	320	12	10	830	$2.09
6. milk	100	8	2.5	125	$0.89
7. salad	320	31	2	123	$2.49

	Calories	Protein	Fat	Sodium
Lower bound	1800	91	0	0
Upper bound	2200	9999	65	1779

For example, having one serving of ice cream and one serving of chicken would not give him enough calories, but having 6 servings of ice cream would, but that would be too much fat. Luke is okay with fractional servings, but he would like to make sure that the number of servings of each item is either zero, or between 1 and 3. (For example, 0 or 1.2 servings of pizza are okay, but 0.2 or 3.2 are not.) Formulate a LP or MIP to help him decide his optimal diet.

a) What is the decision (in words)? (2 points)
How many servings per day of each food.
b) What is the objective (in words)? (2 points)
Minimize cost.

c) Describe any constraint (in words). (2 points)
Any of the following types of constraints would do:

- At least 1800 of calories per day.
- At most 65 of fat per day.
- If a food is included, then the servings must be within 1 and 3.

d) Write a concrete MIP formulation using the correct mathematical notation, specifying the decision variables, the objective function and all constraints. (15 points)
DV:

- $X_1, \cdots, X_7$ for amount of food i in the diet. (continuous)
- $Z_1, \cdots, Z_7$ for whether each food is included at all. (binary)

$$
\begin{aligned}
\text{Min.} \quad & 1.59X_1 + 2.89X_2 + 1.99X_3 + 1.89X_4 + 2.09X_5 + 0.89X_6 + 2.49X_7 \\
\text{s.t.} \quad & \\
& 1800 \leq 330X_1 + 420X_2 + 320X_3 + 380X_4 + 320X_5 + 100X_6 + 320X_7 \leq 2200 \\
& 91 \leq 8X_1 + 32X_2 + 15X_3 + 4X_4 + 12X_5 + 8X_6 + 31X_7 \leq 9999 \\
& 0 \leq 15X_1 + 10X_2 + 20X_3 + 19X_4 + 10X_5 + 2.5X_6 + 2X_7 \leq 65 \\
& 0 \leq 180X_1 + 300X_2 + 820X_3 + 270X_4 + 830X_5 + 125X_6 + 123X_7 \leq 1779 \\
& Z_i \leq X_i \leq 3Z_i \qquad \text{for each food } i \\
& X_i \geq 0
\end{aligned}
$$

e) Define data variables to encode all inputs needed when generalizing the above MIP formulation to arbitrarily large data sets. (4 points)

- I: set of foods.
- J: set of nutrients.
- c_i: cost of food i.
- a_{ij}: amount of nutrient j in one serving of food i.
- L_j: lower bound in daily intake of nutrient j.
- U_j: upper bound of nutrient j.

f) Write an abstract MIP formulation using the data variables you defined in part e), specifying the decision variables, the objective function and all constraints. (10 points)
DV:

- X_i: amount of food i in the diet. (continuous)
- Z_i: whether to include food i at all. (binary)

$$
\begin{aligned}
\text{Min.} \quad & \sum_i c_i X_i \\
\text{s.t.} \quad & \\
& L_j \leq \sum_i a_{ij} X_i \leq U_j \qquad \text{for each nutrient } j. \\
& Z_i \leq X_i \leq 3Z_i \qquad \text{for each food } i. \\
& X_i \geq 0
\end{aligned}
$$

Q8. Gurobi Coding (25 points)

Consider the following LP for optimizing how to best payoff one's credit card debts (assuming that one stops using these cards from now on).

Data:

- I: the set of credit cards you carry.
- n: the number of months to optimize.
- $T = \{0, 1, \cdots, n-1\}$: the set of months.
- r_i: the monthly interest rate of card i.
- B_i: the current balance on credit card i.
- C: the total cash allocated each month to pay the debts.

The set of cards I, current balance B_i and interest rate r_i are stored in the following Excel sheet called `data.xlsx`. The other data (n and C) are supplied in the Python code below.

	A	B	C
1	**Credit Card**	**Balance**	**Monthly Rate**
2	Saks Fifth Avenue	$20,000	0.005
3	Bloomingdale's	$50,000	0.010
4	Macy's	$40,000	0.015

Decision variables:

- x_{it}: how much to pay off credit card $i \in I$ in month $t \in T$.
- y_{it}: the balance at the end of month t on credit card i.

LP:

$$
\begin{array}{rrll}
 & \text{Minimize:} & \sum_{i\in I}\sum_{t\in T} x_{it} + \sum_{i\in I} y_{i(n-1)} & \text{(total payment + final month's balance)} \\
 & \text{subject to:} & & \\
\text{(Maximum payment each month)} & & \sum_{i\in I} x_{it} \le C & \text{for each month } t \in T. \\
\text{(Initial cash flow)} & & (1+r_i)B_i - x_{i0} = y_{i0} & \text{for each card } i. \\
\text{(Cash flow)} & & (1+r_i)y_{i(t-1)} - x_{it} = y_{it} & \text{for each card } i \text{, each month } t \ge 1. \\
\text{(Non-negativity)} & & x_{it}, y_{it} \ge 0 & \text{for all } i, t.
\end{array}
$$

Implement the above LP by completing the following code. You should store the output in a DataFrame named `answer` where the row indices are the months T, and the column names are the names of the credit cards I. In each cell, you should indicate the payment x_{it} to that credit card in that month.

In the below, we use the same shorthands for adding variables with indices as introduced in the solutions to question 4.

```
[21]: import pandas as pd
      data=pd.read_excel('Q8.xlsx',index_col=0)
      n=36
      C=5000

      # Write your Python code below
      import gurobipy as grb
```

```python
mod=grb.Model()

# Prepare indices
I=data.index
T=range(n)

# Define decision variables
x=mod.addVars(I,T,lb=0)
y=mod.addVars(I,T,lb=0)

# Defining objective and constraints, following the formulation line by line
mod.setObjective(sum(x[i,t] for i in I for t in T)+sum(y[i,n-1] for i in I))

for t in T:
    mod.addConstr(sum(x[i,t] for i in I)<=C)

for i in I:
    ri=data.loc[i,'Monthly Rate']
    mod.addConstr((1+ri)*data.loc[i,'Balance']-x[i,0]==y[i,0])
    for t in T[1:]:
        mod.addConstr((1+ri)*y[i,t-1]-x[i,t]==y[i,t])

mod.optimize()
table=[]
for t in T:
    table.append([x[i,t].x for i in I])
answer=pd.DataFrame(table,index=T,columns=I)
answer
```

Credit Card	Saks Fifth Avenue	Bloomingdale's	Macy's
0	0.000000	0.000000	5000.000000
1	0.000000	0.000000	5000.000000
2	0.000000	0.000000	5000.000000
3	0.000000	0.000000	5000.000000
4	0.000000	0.000000	5000.000000
5	0.000000	0.000000	5000.000000
6	0.000000	0.000000	5000.000000
7	0.000000	0.000000	5000.000000
8	0.000000	2061.059449	2938.940551
9	0.000000	5000.000000	0.000000
10	0.000000	5000.000000	0.000000
11	0.000000	5000.000000	0.000000
12	0.000000	5000.000000	0.000000
13	0.000000	5000.000000	0.000000
14	0.000000	5000.000000	0.000000
15	0.000000	5000.000000	0.000000
16	0.000000	5000.000000	0.000000
17	0.000000	5000.000000	0.000000
18	0.000000	5000.000000	0.000000
19	0.000000	5000.000000	0.000000
20	4115.371424	884.628576	0.000000
21	5000.000000	0.000000	0.000000

22	5000.000000	0.000000	0.000000
23	5000.000000	0.000000	0.000000
24	3307.112658	0.000000	0.000000
25	0.000000	0.000000	0.000000
26	0.000000	0.000000	0.000000
27	0.000000	0.000000	0.000000
28	0.000000	0.000000	0.000000
29	0.000000	0.000000	0.000000
30	0.000000	0.000000	0.000000
31	0.000000	0.000000	0.000000
32	0.000000	0.000000	0.000000
33	0.000000	0.000000	0.000000
34	0.000000	0.000000	0.000000
35	0.000000	0.000000	0.000000