42101 - Introduction to Operations Research
Solutions

Christian Keilstrup Ingwersen
Morten Bondorf Gerdes
Richard Lusby
Stefan Røpke

Updated: March 4, 2019

Danmarks Tekniske Universitet

Contents

1 Introduction to Linear Programming

1.1 Exercise 1

Let x_1, x_2, x_3 represent how many liters of each of kind of chocolate that should be produced. The profit of selling the two first products is 200 kr per liter and the profit of selling the last product is 700 kr per liter. This gives us the objective function $Z = 200x_1 + 200x_2 + 700x_3$ which we want to maximize.

Since Peter only has 22 bars of chocolate and premium uses 2 bars, superb 1 bar and elite 3 bars, we have the constraint: $2x_1 + x_2 + 3x_3 \leq 22$.

In order to produce the hot chocolate Peter also needs milk powder. Peter has 20 bags of milk powder, which gives the constraint: $x_1 + 2x_2 + 4x_3 \leq 20$.

All the constraints related to the resource limitations are now defined, but since Peter only is able to carry 10 liters of hot chocolate we need to add a constraint constraining the total production i.e. $x_1 + x_2 + x_3 \leq 10$.

Combining the constraints and the objective function, we have the following LP

$$\max \quad Z = 200x_1 + 200x_2 + 700x_3 \tag{1}$$
$$2x_1 + x_2 + 3x_3 \leq 22 \tag{2}$$
$$x_1 + 2_x 2 + 4x_3 \leq 20 \tag{3}$$
$$x_1 + x_2 + x_3 \leq 10 \tag{4}$$
$$x_1, x_2, x_3 \geq 0 \tag{5}$$

Solving the problem in Julia, we find that the optimal solution is $Z^* = 3640$ kr and Peter should produce 5.6 liters of premium and 3.6 liters of elite hot chocolate.

To solve the problem in Julia, the following code is used

```
using JuMP
using GLPKMathProgInterface
m = Model(solver = GLPKSolverLP())
@variable(m, x1 >= 0 )
@variable(m, x2 >= 0 )
@variable(m, x3 >= 0 )
@objective(m, Max, 200x1 +200x2 + 700x3 )
@constraint(m, x1 + x2 + x3 <= 10 )
@constraint(m, 2x1 + x2 + 3x3 <= 22 )
@constraint(m, x1 + 2x2 + 4x3 <= 20 )
status = solve(m)
println("Objective value: ", getobjectivevalue(m))
println("x1 = ", getvalue(x1))
println("x2 = ", getvalue(x2))
println("x3 = ", getvalue(x3))
```

1.2 Exercise 2

From the given text the use of frame parts and electrical components are seen to be the following for the two products.

Product	Frame parts	Electrical components	Profit
1	1	2	1
2	3	2	2*

Table 1: *Any excess over 60 units of product 2 brings no profit

We now define the following decision variables:
x_1 = production of product 1 (units)
x_2 = production of product 2 (units)

With a profit of \$1 for product 1 and \$2 for product 2, the total profit is $Z = x_1 + 2x_2$, which also is our objective function.
The resources are limited, which gives us a couple of constraints. Since there only are 200 units of frame parts and product 1 uses 1 frame part per unit, while product 2 uses 3 frame parts per unit, we have the following constraint:
$x_1 + 3x_2 \leq 200$.

Likewise since there only are 300 units of electrical components and product 1 and 2 each uses 2 components per unit, we have the following constraint: $2x_1 + 2x_2 \leq 300$.

Since there is no profit if more than 60 units of x_2 are produced, we introduce the constraint $x_2 \leq 60$.

Combined this gives us the following LP

$$max \qquad Z = x_1 + 2x_2 \tag{6}$$
$$x_1 + 3x_2 \leq 200 \tag{7}$$
$$2x_1 + 2_x2 \leq 300 \tag{8}$$
$$x_2 \leq 60 \tag{9}$$
$$x_1, x_2 \geq 0 \tag{10}$$

When using the graphical method, the optimal solution is seen to lie in the intersection of the two first constraints resulting in an objective value of $Z^* = \$175$, where 125 units of product 1 are produced and 25 units of product 2 are produced.

1.3 Exercise 3

The money available for investment is 6000\$ and the time available is 600 hours. The description of the situation can be seen in Table 2.

Table 2: Description of investment problem

	Project 1	Project 2	Limitations
Profit	4500	4500	
Time	400	500	600
Money	5000	4000	6000

Based on the exercise and Table 2 a linear programming model is formulated.
Let x_1 be the investment share in project 1 and let x_2 be the investment share in project 2. The profit of investing in both project 1 and project 2 is 4500 \$. The total profit is then calculated by $Z = 4500x_1 + 4500x_2$. The money needed for investing in project 1 is 5000\$ and for project 2 it is 4000\$. It is not possible to invest more money than you have and therefore you are bounded by 6000\$. This can be formulated as the constraint $5000x_1 + 4000x_2 \leq 6000$.

Since we only have 600 hours available we also need to constrain the time spend on each project i.e. $400x_1 + 500x_2 \leq 600$.
Since both x_1 and x_2 are fractions between 0 and 1 it holds that $x_1 \in [0, 1]$ and $x_2 \in [0, 1]$.
The final model is given by

$$\max \quad Z = 4500x_1 + 4500x_2 \tag{11}$$
$$\text{s.t.} \quad 400x_1 + 500x_2 \leq 600 \tag{12}$$
$$5000x_1 + 4000x_2 \leq 6000 \tag{13}$$
$$x_1, x_2 \in [0, 1] \tag{14}$$

The following plot shows the objective function as the blue line, and the constraints as the red and green line.

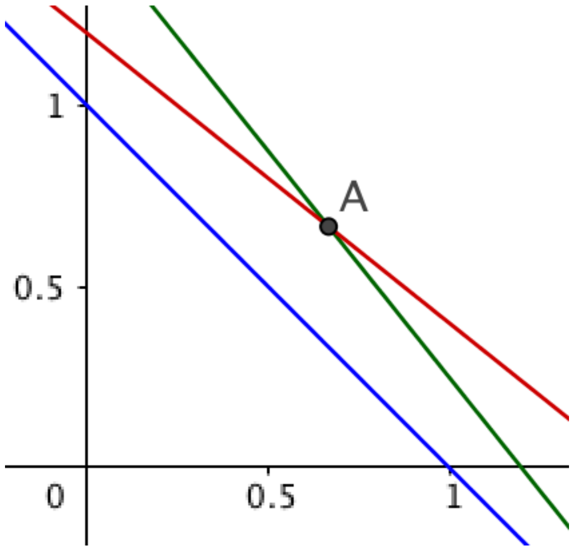


Figure 1: Graphical solution of investment problem

It can be seen that the optimal value is located in the intersection between the two constraints. The optimal value is found by solving two equations with two unknowns.

$$400x_1 + 500x_2 = 600 \textbf{ and } 5000x_1 + 4000x_2 = 6000 \iff \tag{15}$$
$$x_1 = x_2 = \frac{2}{3} \tag{16}$$

Putting the optimal value for the decision variables into the objective function gives the optimal value $Z = 4500 \cdot \frac{2}{3} + 4500 \cdot \frac{2}{3} = 6000$
This means that in order to maximize your profit you should invest $\frac{2}{3}$ of the total investment in project 1 and $\frac{2}{3}$ of the total investment in project 2, which will result in a profit of 6000$
Solving the problem with Julia requires the following piece of code

```
using JuMP
using GLPKMathProgInterface

m = Model(solver = GLPKSolverLP())
@variable(m,0<=x1<=1)
@variable(m,0<=x2<=1)


@objective(m, Max, 4500*x1+4500*x2)
@constraint(m,400*x1+500*x2<=600)
```

```
@constraint(m,5000*x1+4000*x2<=6000)

status = solve(m)

println("Objective value: ",getobjectivevalue(m))
println("x1: ",getvalue(x1))
println("x2: ",getvalue(x2))
```

1.4 Exercise 4

Based on the table in the description a linear program is formulated. The objective is to minimize the cost of the nutrition plan. Let x_1 be the the number of portions of steak per day and let x_2 be the number of portions of potatoes per day. The model can be formulated directly from the table.

$$\min \quad Z = 4x_1 + 2x_2 \tag{17}$$
$$\text{s.t.} \quad 5x_1 + 15x_2 \geq 50 \tag{18}$$
$$20x_1 + 5x_2 \geq 40 \tag{19}$$
$$15x_1 + 2x_2 \leq 60 \tag{20}$$
$$x_1 \geq 0, x_2 \geq 0 \tag{21}$$

The following plot shows the objective function as the orange line, and the constraints as the blue, red and green line.

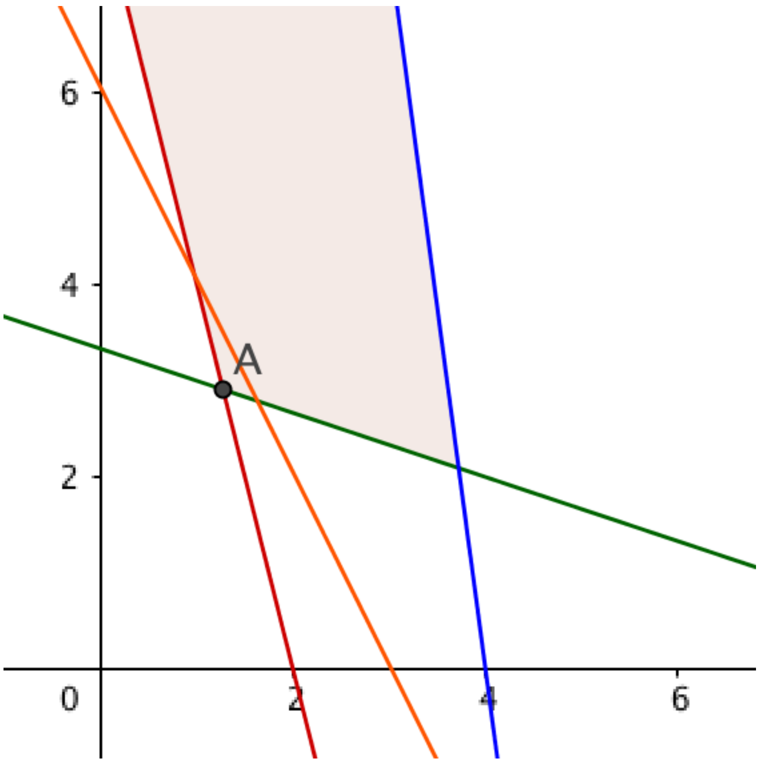


Figure 2: Graphical solution to nutrition problem

It can be seen that the optimal solution is the intersection between the red line and the green line. The green line represents constraint number 1 and the red line represent constraint number 2. The best solution is found by solving two equations with two unknowns.

$$5x_1 + 15x_2 = 50 \textbf{ and } 20x_1 + 5x_2 = 40 \iff \tag{22}$$
$$x_1 = \frac{14}{11}, x_2 = \frac{32}{11} \tag{23}$$

The solution gives the minimum cost $Z^* = \frac{120}{11}$. This means that Ralph has to eat $\frac{14}{11}$ portions of steak a day and $\frac{32}{11}$ portions of potatoes a day which will cost him $\frac{120}{11}$\$
Solving the problem with Julia requires the following piece of code

```
using JuMP
using GLPKMathProgInterface

m = Model(solver = GLPKSolverLP())
@variable(m,x1>=0)
@variable(m,x2>=0)


@objective(m, Min, 4*x1+2*x2)
@constraint(m,5*x1+15*x2>=50)
@constraint(m,20*x1+5*x2>=40)
@constraint(m,15*x1+2*x2<=60)

status = solve(m)

println("Objective value: ",getobjectivevalue(m))
println("x1: ",getvalue(x1))
println("x2: ",getvalue(x2))
```

2 Linear Programming: Simplex method

2.1 Exercise 1

The problem is rewritten in augmented form by introducing slack variables

$$\begin{aligned} \text{Max} \quad & Z - 2x_1 - x_2 = 0 && (24)\\ \text{s.t.} \quad & x_1 + x_2 + x_3 = 40 && (25)\\ & 4x_1 + x_2 + x_4 = 100 && (26)\\ & x_1, x_2, x_3, x_4 \geq 0 && (27) \end{aligned}$$

The first table of the simplex method is then established

	Z	x_1	x_2	x_3	x_4	RHS
Z	1	-2	-1	0	0	0
x_3	0	1	1	1	0	40
x_4	0	4	1	0	1	100

The pivot column is the column with the most negative coefficient in the objective row. In this case it is x_1. The minimum ratio test shows that x_4 is the leaving variable.

	Z	x_1	x_2	x_3	x_4	RHS	ratio
Z	1	-2	-1	0	0	0	
x_3	0	1	1	1	0	40	40
x_4	0	4	1	0	1	100	25

To make the new tabular legal, row 2 (R2) is divided by 4. Row 2 is then added two times to row 0 (R0), and subtracted 1 time from row 1(R1).

	Z	x_1	x_2	x_3	x_4	RHS	ratio
Z	1	0	$-\frac{1}{2}$	0	$\frac{1}{2}$	50	
x_3	0	0	$\frac{3}{4}$	1	-$\frac{1}{4}$	15	20
x_1	0	1	$\frac{1}{4}$	0	$\frac{1}{4}$	25	100

x_3 is the leaving variable while x_2 is the new basis variable.
Operations to make the new tabular legal: $R1_{new} = \frac{R1}{\frac{3}{4}}$, $R0_{new} = R0 + \frac{1}{2} \cdot R1_{new}$, $R2_{new} = R2 - \frac{1}{4} \cdot R1_{new}$

	Z	x_1	x_2	x_3	x_4	RHS
Z	1	0	0	$\frac{2}{3}$	$\frac{1}{3}$	60
x_2	0	0	1	$\frac{4}{3}$	$-\frac{1}{3}$	20
x_1	0	1	0	$-\frac{1}{3}$	$\frac{1}{3}$	20

Since there are no negative coefficients in the objective row (R0), the optimal solution is found. The optimal solution is $x_1 = x_2 = 20$, corresponding to $Z^* = 60$

2.2 Exercise 2

Before starting the simplex method, the problem is first rewritten to augmented form by introducing three slack variables x_4, x_5 and x_6. Rewriting the problem we end up with

$$\begin{aligned} \max \quad & Z - x_1 + 7x_2 - 3x_3 = 0 && (28)\\ \text{s.t.} \quad & 2x_1 + x_2 - x_3 + x_4 = 4 && (29)\\ & 4x_1 - 3x_2 + x_5 = 2 && (30)\\ & -3x_1 + 2x_2 + x_3 + x_6 = 3 && (31)\\ & x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 && (32) \end{aligned}$$

This is used to establish the initial table of the simplex method as seen below.

Bv	Z	x_1	x_2	x_3	x_4	x_5	x_6	Rhs	Ratio
Z	1	-1	7	-3	0	0	0	0	
x_4	0	2	1	-1	1	0	0	4	
x_5	0	4	-3	0	0	1	0	2	
x_6	0	-3	2	1	0	0	1	3	3

Here x_3 is joining basis as it is the variable with the most negative coefficient in the objective row (R0). Since the only variable greater than zero in the x_3 column is in the fourth row, x_6 is the leaving variable. Next step is to ensure the pivot variable is equal to one and the other values in the pivot column should be equal to zero. The pivot variable is already one, so only the following operations are needed:
$R1_{new} = R1 + R3$ and $R0_{new} = R0 + 3 \cdot R3$ resulting in the following table

Bv	Z	x_1	x_2	x_3	x_4	x_5	x_6	Rhs	Ratio
Z	1	-10	13	0	0	0	3	9	
x_4	0	-1	3	0	1	0	1	7	
x_5	0	4	-3	0	0	1	0	2	0.5
x_3	0	-3	2	1	0	0	1	3	

Next up x_1 is entering basis and x_5 is leaving basis. To come to the next iteration, the following operations are performed:
$R2_{new} = R2/4 \rightarrow R1_{new} = R1 + R2_{new} \rightarrow R0_{new} = R0 + 10 \cdot R2_{new} \rightarrow R3_{new} = R3 + 3 \cdot R2_{new}$

Bv	Z	x_1	x_2	x_3	x_4	x_5	x_6	Rhs	Ratio
Z	1	0	5.5	0	0	2.5	3	14	
x_4	0	0	2.25	0	1	0.25	1	7.5	
x_1	0	1	-0.75	0	0	0.25	0	0.5	
x_3	0	0	-0.25	1	0	0.75	1	4.5	

There are no more negative values in the objective row (R0), which means we have found an optimal solution $Z^* = 14$, when $(x_1, x_2, x_3) = (0.5, 0, 4.5)$.

2.3 Exercise 3

x_2 should leave the basis. We only have to consider x_2, x_3, x_4 as these are the ones with coefficient greater than zero 0 in the column of x_1. Of these x_2 has ratio $\frac{0}{3} = 0$ which is minimal.

2.4 Exercise 4

a) The abstract LP model is:

$$\min \sum_{j=1}^{n} c_j x_j \tag{33}$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \quad \forall i = 1, \ldots, m \tag{34}$$

$$x_j \geq 0 \quad \forall j = 1, \ldots, n \tag{35}$$

Let's start by looking at constraints (49). We have a constraint of this type for all $i = 1, \ldots, m$, that is for every time interval that we have defined. Each such constraint ensures that we have enough agents on duty for the particular time interval (remember that a_{ij} is one if shift j covers time interval i and zero otherwise). Constraints (50) are non-negativity requirements. We could add that all variables should take integer values, but if we do so, our model is no longer a LP.
The objective function (48) minimizes the cost of assigning agents to shifts. Each term $c_j x_j$ in the summation computes the cost of assigning x_j agents to shift j.
b,c)

```
using JuMP
using GLPKMathProgInterface

# This is the data defined in task (c)
m = 3
n = 5
b = [2,4,3]
c = [1500, 2000, 800, 1800, 1300]
a = [1 1 0 0 0; 0 1 1 1 0; 0 0 0 1 1]

# This is the Julia implementation of the abstract model
model = Model(solver = GLPKSolverLP())
@variable(model, x[1:n] >= 0 )
@objective(model, Min, sum(c[j]*x[j] for j=1:n) )
@constraint(model, [i=1:m], sum(a[i,j]*x[j] for j=1:n) >= b[i])

status = solve(model)
println("Objective value: ", getobjectivevalue(model))
println("x = ", getvalue(x))
```

d) When using the data from (week2-ex3-dat.txt) one gets an objective value of 21250 and a solution
x = [2, 0, 0, 0, 0, 0, 1, 1, 4, 3, 0, 1, 2, 2, 1, 1, 3, 0, 0, 0, 0, 0, 0, 0]

2.5 Exercise 5

The problem is rewritten only with positive variables by introducing: $x_1' = x_1 + 7$

$$\max \qquad Z = x_1' + 3x_2 - 7 \tag{36}$$

$$\text{s.t.} \qquad x_1' + 2x_2 \leq 7 \tag{37}$$

$$x_1', x_2 \geq 0 \tag{38}$$

We write this LP on augmented form and insert in a tableau. We pivot x_2 into the basis and then the optimality test show that we are done.

```
-----------------------------------------------
b.v. | eq. |    x1'      x2       x3   |   RHS
-----------------------------------------------
     |   0 |  -1.00    -3.00     0.00  |  -7.00
  x3 |   1 |   1.00     2.00     1.00  |   7.00
-----------------------------------------------
b.v. | eq. |    x1'      x2       x3   |   RHS
-----------------------------------------------
     |   0 |   0.50     0.00     1.50  |   3.50
  x2 |   1 |   0.50     1.00     0.50  |   3.50
```

This gives us the solution $x_1' = 0, x_2 = 3.5$ and $Z = 3.5$ but remember that this is the solution to the rewritten problem. The solution to the original problem is:

$$x_1 = x_1' - 7 = 0 - 7 = -7, x_2 = 3.5, Z = 3.5$$

2.6 Exercise 6

Since x_2 is unrestricted we write it as $x_2 = x_2^+ - x_2^-$ with $x_2^+ \geq 0$ and $x_2^- \geq 0$ and substitute this into the LP:

$$\min Z = x_1 + 2\left(x_2^+ - x_2^-\right)$$

subject to

$$\begin{aligned} -x_1 - \left(x_2^+ - x_2^-\right) &\leq 5 \\ 2x_1 + \left(x_2^+ - x_2^-\right) &\leq 6 \\ x_1, x_2^+, x_2^- &\geq 0 \end{aligned}$$

We have to remember to change minimization to maximization and gets

$$\max -Z = -x_1 - 2x_2^+ + 2x_2^-$$

subject to

$$\begin{aligned} -x_1 - x_2^+ + x_2^- &\leq 5 \\ 2x_1 + x_2^+ - x_2^- &\leq 6 \\ x_1, x_2^+, x_2^- &\geq 0 \end{aligned}$$

We then solve this LP using the simplex algorithm:

```
------------------------------------------------------------------
b.v.  | eq.  |     x1       x2+       x2-        x3        x4   |    RHS
------------------------------------------------------------------
      |   0  |   1.00      2.00     -2.00      0.00      0.00   |   0.00
  x3  |   1  |  -1.00     -1.00      1.00      1.00      0.00   |   5.00
  x4  |   2  |   2.00      1.00     -1.00      0.00      1.00   |   6.00
------------------------------------------------------------------
b.v.  | eq.  |     x1       x2+       x2-        x3        x4   |    RHS
------------------------------------------------------------------
      |   0  |  -1.00      0.00      0.00      2.00      0.00   |  10.00
 x2-  |   1  |  -1.00     -1.00      1.00      1.00      0.00   |   5.00
  x4  |   2  |   1.00      0.00      0.00      1.00      1.00   |  11.00
------------------------------------------------------------------
b.v.  | eq.  |     x1       x2+       x2-        x3        x4   |    RHS
------------------------------------------------------------------
      |   0  |   0.00      0.00      0.00      3.00      1.00   |  21.00
 x2-  |   1  |   0.00     -1.00      1.00      2.00      1.00   |  16.00
  x1  |   2  |   1.00      0.00      0.00      1.00      1.00   |  11.00
```

We get the solution $x_1 = 11, x_2^- = 16$ and $-Z^* = 21 \Rightarrow Z^* = -21$. We rewrite to the original variables and get the solution $x_1 = 11, x_2 = x_2^+ - x_2^- = -16, Z^* = -21$

d) Now we have to solve

$$\min x_1 + 2x_2$$

subject to

$$\begin{aligned} -x_1 - x_2 &\leq 5 \\ x_1 + 2x_2 &\leq 6 \\ x_1 &\geq 0 \\ x_2 &\in \mathbb{R} \end{aligned}$$

In the same way as before we re-write this to

$$\min Z = x_1 + 2\left(x_2^+ - x_2^-\right)$$

subject to

$$-x_1 - \left(x_2^+ - x_2^-\right) \leq 5$$
$$x_1 + 2\left(x_2^+ - x_2^-\right) \leq 6$$
$$x_1, x_2^+, x_2^- \geq 0$$

changing to maximization:

$$\max -Z = -x_1 - 2x_2^+ + 2x_2^-$$

subject to

$$-x_1 - x_2^+ + x_2^- \leq 5$$
$$x_1 + 2x_2^+ - 2x_2^- \leq 6$$
$$x_1, x_2^+, x_2^- \geq 0$$

We solve this using the simplex algorithm:

```
-----------------------------------------------------------------
b.v. | eq. |     x1       x2+       x2-        x3        x4  |   RHS
-----------------------------------------------------------------
     |   0 |   1.00      2.00     -2.00      0.00      0.00  |   0.00
  x3 |   1 |  -1.00     -1.00      1.00      1.00      0.00  |   5.00
  x4 |   2 |   1.00      2.00     -2.00      0.00      1.00  |   6.00
-----------------------------------------------------------------
b.v. | eq. |     x1       x2+       x2-        x3        x4  |   RHS
-----------------------------------------------------------------
     |   0 |  -1.00      0.00      0.00      2.00      0.00  |  10.00
 x2- |   1 |  -1.00     -1.00      1.00      1.00      0.00  |   5.00
  x4 |   2 |  -1.00      0.00      0.00      2.00      1.00  |  16.00
```

we notice that x_1 should enter the basis in the last tableau. However, since we have only negative elements in the column it means that increasing x_1 would cause $-Z$ to increase as well as the two basis variables x_2^- and x_4. This implies that the problem is unbounded, meaning that Z goes to $-\infty$.

3 Linear Programming: Two phase method, basic graph theory

3.1 Exercise 1

The problem is rewritten in augmented form by introducing artificial variables and surplus variables

$$\begin{aligned} \max \quad & -Z + 2x_1 + x_2 + 3x_3 = 0 && (39)\\ \text{s.t.} \quad & 5x_1 + 2x_2 + 7x_3 + \overline{x_4} = 420 && (40)\\ & 3x_1 + 2x_2 + 5x_3 - x_5 + \overline{x_6} = 280 && (41)\\ & x_1, x_2, x_3, \overline{x_4}, x_5, \overline{x_6} \geq 0 && (42) \end{aligned}$$

Since the problem has artificial variables, the two-phase method is used. First step is to minimize the artificial variables: min $\quad Z = \overline{x_4} + \overline{x_6} \quad \iff \quad$ max $\quad -Z + \overline{x_4} + \overline{x_6}=0$
The constraints remain the same as in the original problem.
The first tabular is

	Z	x_1	x_2	x_3	$\overline{x_4}$	x_5	$\overline{x_6}$	RHS
Z	-1	0	0	0	1	0	1	0
$\overline{x_4}$	0	5	2	7	1	0	0	420
$\overline{x_6}$	0	3	2	5	0	-1	1	280

The tabular is not legal, since the coefficient in R0 is 1 for both $\overline{x_4}$ and $\overline{x_6}$. Subtracting R1 and R2 from R0 gives the following tabular

	Z	x_1	x_2	x_3	$\overline{x_4}$	x_5	$\overline{x_6}$	RHS	Ratio
Z	-1	-8	-4	-12	0	1	0	-700	
$\overline{x_4}$	0	5	2	7	1	0	0	420	60
$\overline{x_6}$	0	3	2	5	0	-1	1	280	56

$\overline{x_6}$ is the leaving variable and x_3 is the entering variable.
Row operations: $R2_{new} = \frac{R2}{5}$, $R0_{new} = R0 + 12 \cdot R2_{new}$, $R1_{new} = R1 - 7 \cdot R2_{new}$

	Z	x_1	x_2	x_3	$\overline{x_4}$	x_5	$\overline{x_6}$	RHS	Ratio
Z	-1	-$\frac{4}{5}$	$\frac{4}{5}$	0	0	-$\frac{7}{5}$	$\frac{12}{5}$	-28	
$\overline{x_4}$	0	$\frac{4}{5}$	-$\frac{4}{5}$	0	1	$\frac{7}{5}$	-$\frac{7}{5}$	28	20
x_3	0	$\frac{3}{5}$	$\frac{2}{5}$	1	0	-$\frac{1}{5}$	$\frac{1}{5}$	56	-280

$\overline{x_4}$ is the leaving variable and x_5 is the entering variable.
Row operations: $R1_{new} = \frac{R1}{\frac{7}{5}}$, $R0_{new} = R0 + \frac{7}{5} \cdot R1_{new}$, $R2_{new} = R2 + \frac{1}{5} \cdot R1_{new}$

	Z	x_1	x_2	x_3	$\overline{x_4}$	x_5	$\overline{x_6}$	RHS
Z	-1	0	0	0	1	0	1	0
x_5	0	$\frac{4}{7}$	-$\frac{4}{7}$	0	$\frac{5}{7}$	1	-1	20
x_3	0	$\frac{5}{7}$	$\frac{2}{7}$	1	$\frac{1}{7}$	0	0	60

Phase 1 (minimizing the artificial variables) is now done, since there are no negative coefficients in the objective row.
The original objective function is inserted to the last tabular from phase 1, and the columns of the artificial variables are deleted.

	Z	x_1	x_2	x_3	x_5	RHS
Z	-1	2	1	3	0	0
x_5	0	$\frac{4}{7}$	-$\frac{4}{7}$	0	1	20
x_3	0	$\frac{5}{7}$	$\frac{2}{7}$	1	0	60

First of all the tabular is made legal by subtracting R2 from R0 three times.

	Z	x_1	x_2	x_3	x_5	RHS	Ratio
Z	-1	$-\frac{1}{7}$	$\frac{1}{7}$	0	0	-180	
x_5	0	$\frac{4}{7}$	$-\frac{4}{7}$	0	1	20	35
x_3	0	$\frac{5}{7}$	$\frac{2}{7}$	1	0	60	84

x_5 is the leaving variable and x_1 is the entering variable.
Row operations: $R1_{new} = \frac{R1}{\frac{4}{7}}$, $R0_{new} = R0 + \frac{1}{7} \cdot R1_{new}$, $R2_{new} = R2 - \frac{5}{7} \cdot R1_{new}$

	Z	x_1	x_2	x_3	x_5	RHS
Z	-1	0	0	0	$\frac{1}{4}$	-175
x_1	0	1	-1	0	$\frac{7}{4}$	35
x_3	0	0	1	1	$-\frac{5}{4}$	35

There are no negative coefficients in the objective row and the optimal solution is found. $x_1 = x_3 = 35$ and $Z = 175$.
Since the coefficient of x_2 in the objective row is 0 and x_2 is not a basis variable, there exist more than one optimal solution. Forcing x_2 to be a basis variable, and doing one more iteration of the simplex method, would result in a solution with the same objective value $Z = 175$ but with $x_1 = 70$ and $x_2 = 35$. Linear combinations of the to solutions would result in the same optimal solution: $X^* = w_1 \cdot (35, 0, 35) + w_2 \cdot (70, 35, 0)$, $w_1 + w_2 = 1$ and $w_1, w_2 \geq 0$

3.2 Exercise 2

To rewrite the problem to augmented form we need to introduce a slack variable, a surplus variable and 2 artificial variables. The slack variable (lets say x_4) belongs to the first $\leq$ constraint. One of the artificial variables ($\overline{x_5}$) belongs to the $=$ constraint and the last artificial variable ($\overline{x_6}$) and surplus variable (x_7) belong to the last $\geq$ constraint.
The problem on augmented form is:

$$\text{Max} \quad Z - 2x_1 - 4x_2 - 7x_3 = 0 \tag{43}$$
$$\text{s.t.} \quad 4x_1 + 2x_2 + 2x_3 + x_4 = 60 \tag{44}$$
$$x_1 + 2x_2 + 2x_3 + \overline{x_5} = 30 \tag{45}$$
$$x_1 + 4x_2 + x_3 + \overline{x_6} - x_7 = 40 \tag{46}$$
$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \tag{47}$$

To solve this problem we are going to use the two-phase method. The first phase in the two-phase method is to force the artificial variables to zero, i.e

$$\text{Min} \qquad Z = \overline{x_5} + \overline{x_6}$$

which in augmented form becomes

$$\text{Max} \qquad -Z + \overline{x_5} + \overline{x_6} = 0$$

and is the objective function we are using in the first phase.

The initial table of the first phase of the two-phase method is then:

Bv	Z	x_1	x_2	x_3	x_4	$\overline{x_5}$	$\overline{x_6}$	x_7	Rhs	Ratio
Z	-1	0	0	0	0	1	1	0	0	
x_4	0	4	2	2	1	0	0	0	60	
$\overline{x_5}$	0	1	2	2	0	1	0	0	30	
$\overline{x_6}$	0	1	4	1	0	0	1	-1	40	

However, this table is not in a legal state since the values of the basis variables $\overline{x_5}$ and $\overline{x_6}$ isn't zero in the objective row. Before starting the simplex method we need to make the table legitimate. First step is $R0 - R2$, resulting in

Bv	Z	x_1	x_2	x_3	x_4	$\overline{x_5}$	$\overline{x_6}$	x_7	Rhs	Ratio
Z	-1	-1	-2	-2	0	0	1	0	-30	
x_4	0	4	2	2	1	0	0	0	60	
$\overline{x_5}$	0	1	2	2	0	1	0	0	30	
$\overline{x_6}$	0	1	4	1	0	0	1	-1	40	

Since $\overline{x_6}$ still is one in the objective row while in basis, the table isn't legitimate yet. Next step is $R0 - R3$

Bv	Z	x_1	x_2	x_3	x_4	$\overline{x_5}$	$\overline{x_6}$	x_7	Rhs	Ratio
Z	-1	-2	-6	-3	0	0	0	1	-70	
x_4	0	4	2	2	1	0	0	0	60	30
$\overline{x_5}$	0	1	2	2	0	1	0	0	30	15
$\overline{x_6}$	0	1	4	1	0	0	1	-1	40	10

The table is now legitimate and we can start minimizing the artificial variables. x_2 is entering basis and $\overline{x_6}$ is leaving basis. Next to come to the next iteration, the following operations have been done.
$R3_{new} = R3/4 \to R2_{new} = R2 - 2 \cdot R3_{new} \to R1_{new} = R1 - 2 \cdot R3_{new} \to R0_{new} = R0 + 6 \cdot R3_{new}$

Bv	Z	x_1	x_2	x_3	x_4	$\overline{x_5}$	$\overline{x_6}$	x_7	Rhs	Ratio
Z	-1	-0.5	0	-1.5	0	0	1.5	-0.5	-10	
x_4	0	3.5	0	1.5	1	0	-0.5	0-5	40	26.67
$\overline{x_5}$	0	0.5	0	1.5	0	1	-0.5	0.5	10	6.67
x_2	0	0.25	1	0.25	0	0	0.25	-0.25	10	40

Next up x_3 is entering basis and $\overline{x_6}$ is leaving basis. For next iteration the following operations are done
$R2_{new} = R2/1.5 \to R1_{new} = R1 - 1.5 \cdot R2_{new} \to R0_{new} = R0 + 1.5 \cdot R2_{new} \to R3_{new} = R3 - 1/4 \cdot R2_{new}$

Bv	Z	x_1	x_2	x_3	x_4	$\overline{x_5}$	$\overline{x_6}$	x_7	Rhs	Ratio
Z	-1	0	0	0	0	1	1	0	0	
x_4	0	3	0	0	1	-1	0	0	30	
x_3	0	0.33	0	1	0	0.67	-0.33	0.33	6.67	
x_2	0	0.17	1	0	0	-0.17	0.33	-0.33	8.33	

Since there is no more negative values in the objective row, we have found an optimal solution to the phase one problem, which is a legal solution to the original problem. For the second phase we'll replace the objective row with the original objective function and remove the two columns containing artificial variables.

Bv	Z	x_1	x_2	x_3	x_4	x_7	Rhs	Ratio
Z	1	-2	-4	-7	0	0	0	
x_4	0	3	0	0	1	0	30	
x_3	0	0.33	0	1	0	0.33	6.67	
x_2	0	0.17	1	0	0	-0.33	8.33	

The table is currently not in a legal form since x_2 and x_3 is in basis without having a zero in the objective row. To restore proper form the following operations are done:
$R0_{new} = R0 + 4 \cdot R3$

Bv	Z	x_1	x_2	x_3	x_4	x_7	Rhs	Ratio
Z	1	-1.33	0	-7	0	-1.33	33.33	
x_4	0	3	0	0	1	0	30	
x_3	0	0.33	0	1	0	0.33	6.67	
x_2	0	0.17	1	0	0	-0.33	8.33	

Followed by $R0_{new} = R0 + 7 \cdot R2$

Table 3: My caption

Bv	Z	x_1	x_2	x_3	x_4	x_7	Rhs	Ratio
Z	1	1	0	0	0	1	80	
x_4	0	3	0	0	1	0	30	
x_3	0	0.33	0	1	0	0.33	6.67	
x_2	0	0.17	1	0	0	-0.33	8.33	

Since there are no more negative values in the objective row, we have found an optimal solution at $Z^* = 80$, when $(x_1, x_2, x_3) = (0, 8.33, 6.67)$.

3.3 Exercise 3

a) The abstract LP model is:

$$\min \sum_{j=1}^{n} c_j x_j \tag{48}$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \geq b_i \quad \forall i = 1, \ldots, m \tag{49}$$

$$x_j \geq 0 \quad \forall j = 1, \ldots, n \tag{50}$$

Let's start by looking at constraints (49). We have a constraint of this type for all $i = 1, \ldots, m$, that is for every time interval that we have defined. Each such constraint ensures that we have enough agents on duty for the particular time interval (remember that a_{ij} is one if shift j covers time interval i and zero otherwise). Constraints (50) are non-negativity requirements. We could add that all variables should take integer values, but if we do so, our model is no longer a LP.
The objective function (48) minimizes the cost of assigning agents to shifts. Each term $c_j x_j$ in the summation computes the cost of assigning x_j agents to shift j.
b,c)

```
using JuMP
using GLPKMathProgInterface

# This is the data defined in task (c)
m = 3
n = 5
b = [2,4,3]
c = [1500, 2000, 800, 1800, 1300]
a = [1 1 0 0 0; 0 1 1 1 0; 0 0 0 1 1]

# This is the Julia implementation of the abstract model
model = Model(solver = GLPKSolverLP())
```

```
@variable(model, x[1:n] >= 0 )
@objective(model, Min, sum(c[j]*x[j] for j=1:n) )
@constraint(model, [i=1:m], sum(a[i,j]*x[j] for j=1:n) >= b[i])

status = solve(model)
println("Objective value: ", getobjectivevalue(model))
println("x = ", getvalue(x))
```

d) When using the data from (week2-ex3-dat.txt) one gets an objective value of 21250 and a solution x = [2, 0, 0, 0, 0, 0, 1, 1, 4, 3, 0, 1, 2, 2, 1, 1, 3, 0, 0, 0, 0, 0, 0, 0]

3.4 Exercise 4

Julia model

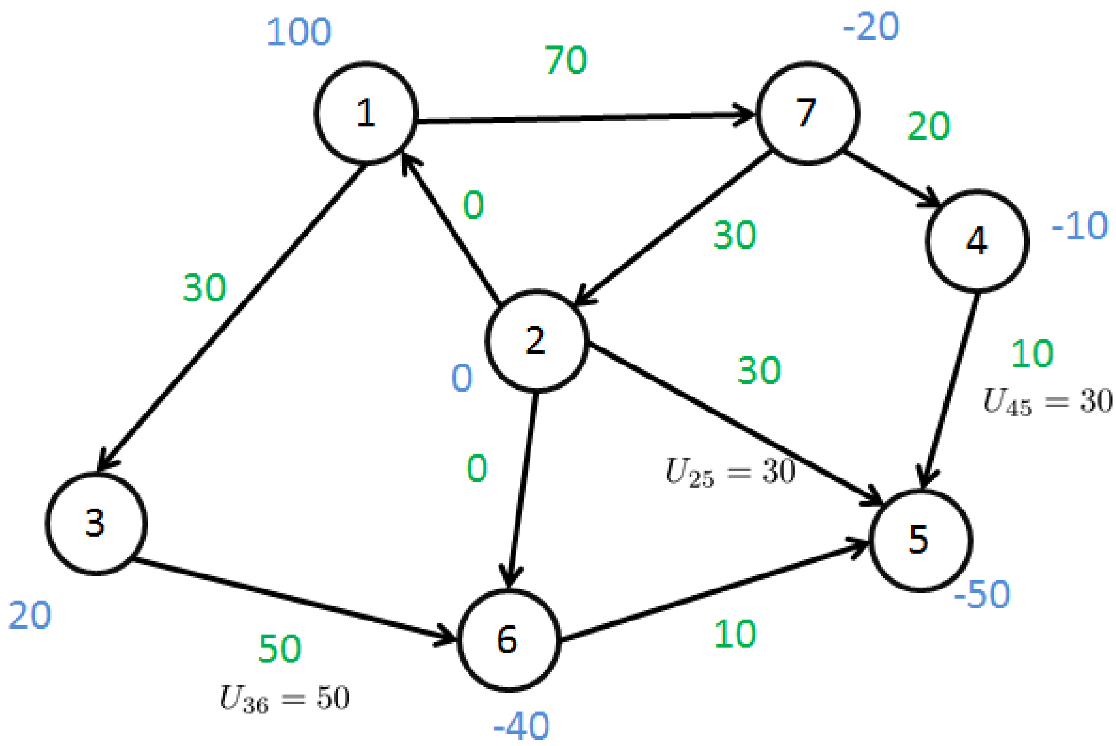
```
using JuMP
using GLPKMathProgInterface
mutable struct Arc
    from::Int64
    to::Int64
    cost::Int64
    UB::Int64
end
m = 7
arcs = [
  Arc(1,3,1,0),Arc(1,7,9,0),Arc(2,1,4,0),
  Arc(2,5,6,30),Arc(2,6,5,0),Arc(3,6,7,50),
  Arc(4,5,8,30),Arc(6,5,12,0),Arc(7,2,2,0),Arc(7,4,3,0)]
demands = [100,0,20,-10,-50,-40,-20]
model = Model(solver = GLPKSolverLP())
@variable(model, x[arcs] >= 0 )
@objective(model, Min, sum(a.cost*x[a] for a in arcs) )
@constraint(model, [i=1:m], sum(x[a] for a in arcs if a.from==i)
                            - sum(x[a] for a in arcs if a.to==i) == demands[i] )
for a in arcs
    if a.UB > 0
        @constraint(model, x[a] <= a.UB )
    end
end
print(model)
status = solve(model)
println("Objective value: ", getobjectivevalue(model))
println("x = ", getvalue(x))
```

Solution:

```
Objective value: 1510.0
[ Arc(1, 3, 1, 0)] = 30.0
[ Arc(1, 7, 9, 0)] = 70.0
[ Arc(2, 1, 4, 0)] = 0.0
[Arc(2, 5, 6, 30)] = 30.0
[ Arc(2, 6, 5, 0)] = 0.0
[Arc(3, 6, 7, 50)] = 50.0
[Arc(4, 5, 8, 30)] = 10.0
[Arc(6, 5, 12, 0)] = 10.0
```

```
[ Arc(7, 2, 2, 0)] = 30.0
[ Arc(7, 4, 3, 0)] = 20.0
```

Drawing
Green numbers are the assigned flow (found by LP), blue numbers are surplus / deficit. We see that the flow respects upper bounds and that the flow from/to each node match with the surplus / deficit.



Modified model. We need to send at least as many units through arc (6,5) at as through arc (2,5). We can express this as

$$x_{65} \geq x_{25}$$

or

$$x_{65} - x_{25} \geq 0$$

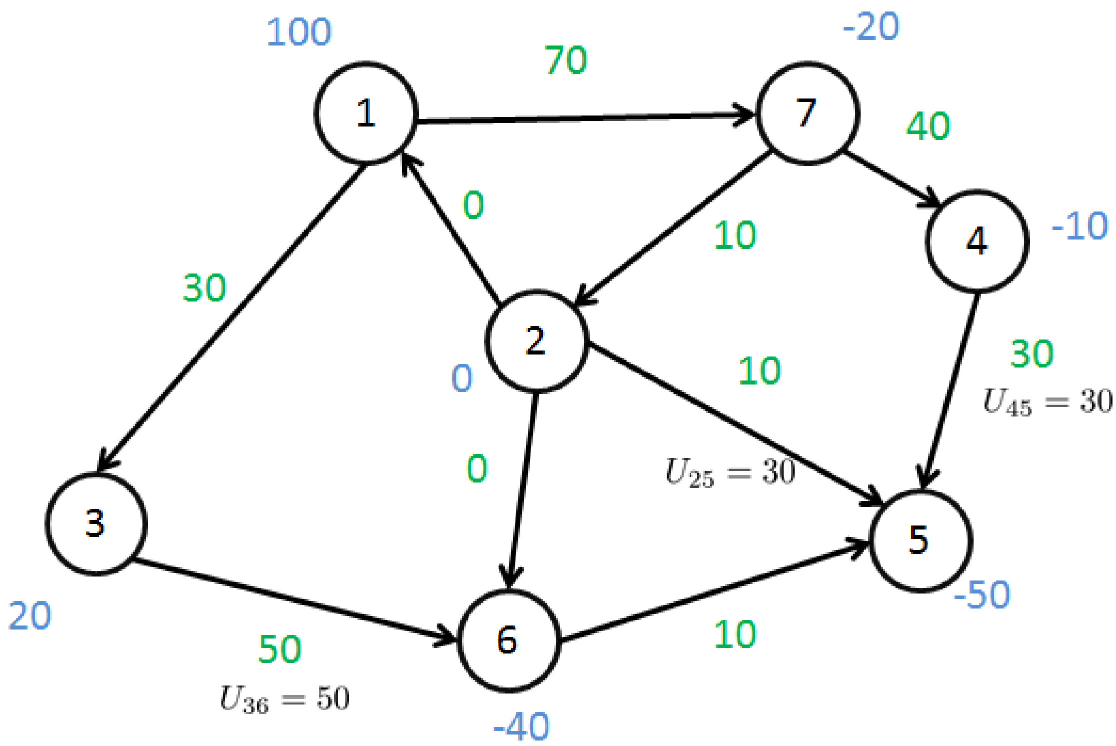
We can write this in Julia as

```
@constraint(model, sum(x[a] for a in arcs if a.from==6 && a.to==5) -
                   sum(x[a] for a in arcs if a.from==2 && a.to==5) >= 0)
```

or

```
@constraint(model, x[arcs[8]] - x[arcs[4]] >= 0)
```

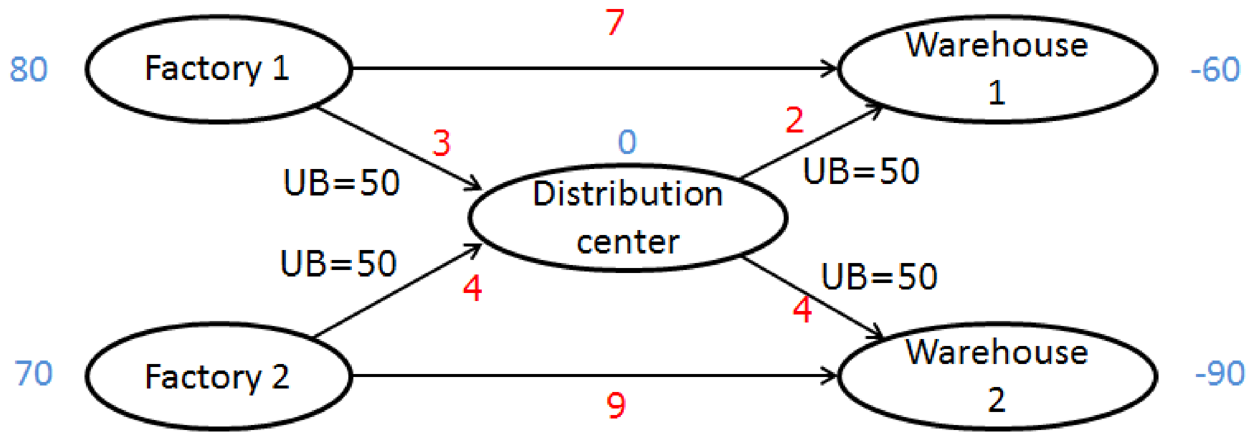
In the second approach we used that arc (6,5) is in the eight position in the list of arcs and arc (2,5) is in the fourth position in the list. With this change the objective value increases to 1570 and the optimal solution becomes:



We see that the added constraint has changed the solution and that the new solution respects the new constraint.

3.5 Exercise 5

The network representation of the problem is: Green numbers are the assigned flow (found by LP), blue numbers are surplus / deficit



We will denote nodes as follows:

	Node #
Factory 1	1
Factory 2	2
Distribution Center	3
Warehouse 1	4
Warehouse 2	5

With this the LP formulation becomes

$$\min 3x_{13} + 7x_{14} + 4x_{23} + 9x_{25} + 2x_{34} + 4x_{35}$$

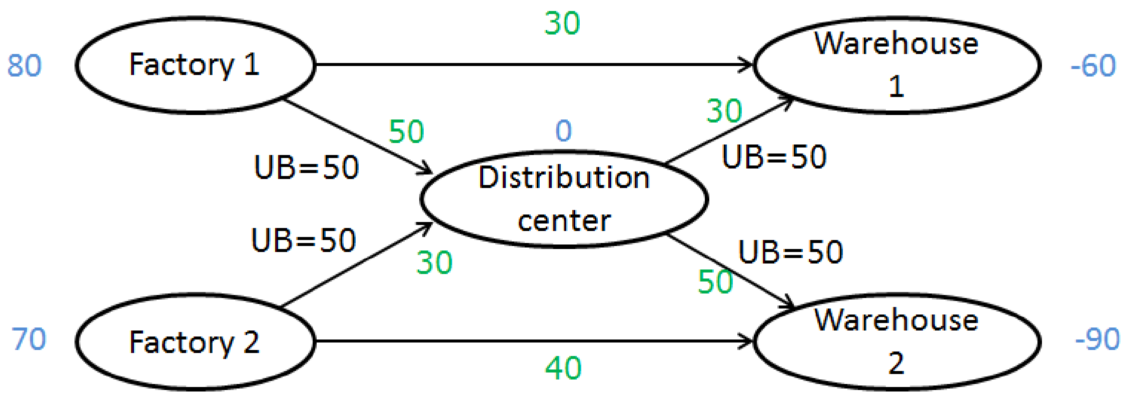
subject to

$$\begin{aligned}
x_{13} + x_{14} &= 80 \\
x_{23} + x_{25} &= 70 \\
-x_{13} - x_{23} + x_{34} + x_{35} &= 0 \\
-x_{14} - x_{34} &= -60 \\
-x_{25} - x_{35} &= -90 \\
x_{14}, x_{25} &\geq 0 \\
0 \leq x_{13}, x_{23}, x_{34}, x_{35} &\leq 50
\end{aligned}$$

We can solve this in Julia by typing in the LP as it is written above, or we can use our generic minimum cost flow model. We use the latter.

```
using JuMP
using GLPKMathProgInterface
type Arc
    from::Int64
    to::Int64
    cost::Int64
    UB::Int64
end
m = 5
arcs = [
  Arc(1,3,3,50),Arc(1,4,7,0),Arc(2,3,4,50),
  Arc(2,5,9,0),Arc(3,4,2,50),Arc(3,5,4,50)]
demands = [80,70,0,-60,-90]
model = Model(solver = GLPKSolverLP())
@variable(model, x[arcs] >= 0 )
@objective(model, Min, sum(a.cost*x[a] for a in arcs) )
@constraint(model, [i=1:m],
  sum(x[a] for a in arcs if a.from==i) -
  sum(x[a] for a in arcs if a.to==i) == demands[i] )
for a in arcs
    if a.UB > 0
        @constraint(model, x[a] <= a.UB )
    end
end
print(model)
status = solve(model)
println("Objective value: ", getobjectivevalue(model))
println("x = ", getvalue(x))
```

When solving this we get an objective of 1100. The flow is shown on the figure below:



80
Factory 1
30
Warehouse 1
-60
50
UB=50
0
30
UB=50
Distribution center
UB=50
30
UB=50
50
70
Factory 2
Warehouse 2
-90
40

4 Linear Programming: Simplex on matrix form

4.1 Exercise 1

From the initial problem, we can construct c, A, B, b and c_B .

$c = \begin{bmatrix} 5 & 8 & 7 & 4 & 6 \end{bmatrix}$, $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $b = \begin{bmatrix} 20 \\ 30 \end{bmatrix}$ $A = \begin{bmatrix} 2 & 3 & 3 & 2 & 2 \\ 3 & 5 & 4 & 2 & 4 \end{bmatrix}$, $c_B = \begin{bmatrix} 0 & 0 \end{bmatrix}$

Further more we want to construct a basis vector for keeping track on the current variables in basis, $x_b = \begin{bmatrix} x_6 \\ x_7 \end{bmatrix}$

When using the simplex method on matrix form, it is important to notice that after each iteration only B and c_B are updated while c, b and A remains constant.

First step is the optimality test: $c_B B^{-1} A - c = \begin{bmatrix} -5 & -8 & -7 & -4 & -6 \end{bmatrix}$
Since -8 is the most negative number, x_2 is entering the basis.

Next step is to find the variable leaving basis, which is done by the minimum ratio test. The right hand side is computed by

$B^{-1}b = \begin{bmatrix} 20 \\ 30 \end{bmatrix}$

And the column under the entering variable is found by

$B^{-1}A_2 = \begin{bmatrix} 3 \\ 5 \end{bmatrix}$, where the subscript denotes the second column of the A matrix.

since $30/5 < 20/3$, x_7 is leaving the basis.

Before beginning the next iteration, B, c_B and x_b are updated.

$x_b = \begin{bmatrix} x_6 \\ x_2 \end{bmatrix}$

Since x_7 is the leaving variable, the column corresponding to x_7 in the B matrix is updated to the column in the A matrix corresponding to the in going variable x_2.

$B = \begin{bmatrix} 1 & 3 \\ 0 & 5 \end{bmatrix}$

The value corresponding to x_7 in c_B is also replaced by the value corresponding to x_2 in the c vector.

$c_B = \begin{bmatrix} 0 & 8 \end{bmatrix}$

Now we are ready for the second iteration, which is exactly like the first iteration. First the optimality test:
$c_B B^{-1} A - c = \begin{bmatrix} -1/5 & 0 & -3/5 & -4/5 & 2/5 \end{bmatrix}$

In this iteration x_4 is seen to enter basis. To find out which variable is leaving the basis, we are using the min ratio test. The right hand side is now

$x_b = \begin{bmatrix} x_6 \\ x_2 \end{bmatrix} = B^{-1}b = \begin{bmatrix} 2 \\ 6 \end{bmatrix}$

And the column under the entering variable is found by

$B^{-1}A_4 = \begin{bmatrix} 4/5 \\ 2/5 \end{bmatrix}$

The ratios are $2/\frac{4}{5} = \frac{5}{2}$ and $6/\frac{2}{5} = 15$, so x_6 is leaving the basis. Again before beginning the next iteration, B, c_B and x_b are updated.

$x_b = \begin{bmatrix} x_4 \\ x_2 \end{bmatrix}$

Since x_6 is the leaving variable, the column corresponding to x_6 in the B matrix is updated to the column in the A matrix corresponding to the in going variable x_4.

$B = \begin{bmatrix} 2 & 3 \\ 2 & 5 \end{bmatrix}$

The value corresponding to x_6 in c_B is also replaced by the value corresponding to x_4 in the c vector.

$c_B = \begin{bmatrix} 4 & 8 \end{bmatrix}$

We are now ready for the third iteration, starting with the optimality test:

$c_B B^{-1} A - c = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

Since there are no negative values, the optimal solution is found. The right hand side in the optimal solution is

$\begin{bmatrix} x_4 \\ x_2 \end{bmatrix} = B^{-1}b = \begin{bmatrix} 5/2 \\ 5 \end{bmatrix}$

And the optimal object value is

$Z^* = c_B B^{-1} b = 50$

4.2 Exercise 2

From the initial problem we know that

$c = \begin{bmatrix} 6 & 1 & 2 \end{bmatrix}$, $b = \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix}$ and $A = \begin{bmatrix} 2 & 2 & 1/2 \\ -4 & -2 & -3/2 \\ 1 & 2 & 1/2 \end{bmatrix}$

And from the final optimal table we can see that $y^* = \begin{bmatrix} 2 & 0 & 2 \end{bmatrix}$ and $S^* = \begin{bmatrix} 1 & 1 & 2 \\ -2 & 0 & 4 \\ 1 & 0 & -1 \end{bmatrix}$

So in order to get to get the full last table, we are missing Z^*, b^*, A^* and $z^* - c$

From our formulas we know that

$Z^* = c_B B^{-1} b = y^* b = 6$

and

$b^* = B^{-1} b = S^* b = \begin{bmatrix} 7 \\ 0 \\ 1 \end{bmatrix}$

Similar for A^*

$$A^* = B^{-1}A = S^*A = \begin{bmatrix} 0 & 4 & 0 \\ 0 & 4 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

and at last

$$z^* - c = c_B B^{-1}A - c = y^*A - c = \begin{bmatrix} 0 & 7 & 0 \end{bmatrix}$$

By using the formulas on the slides, all the missing values are found.

4.3 Exercise 3

From the initial problem, A is determined:

$$A = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 3 \end{bmatrix}$$

From the final simplex tableau, $c_b B^{-1}$ can be read:

$$c_b B^{-1} = \begin{bmatrix} \frac{3}{5} & \frac{4}{5} \end{bmatrix}$$

From fundamental insight we now that:

$$c_b B^{-1}A - c = \begin{bmatrix} \frac{7}{10} & 0 & 0 \end{bmatrix} \Leftrightarrow$$
$$c = c_b B^{-1}A - \begin{bmatrix} \frac{7}{10} & 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{3}{5} & \frac{4}{5} \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 1 & 3 \end{bmatrix} - \begin{bmatrix} \frac{7}{10} & 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{3}{2} & 2 & 3 \end{bmatrix}$$

This means that $c_1 = \frac{3}{2}$, $c_2 = 2$ and $c_3 = 3$
From fundamental insight we know that the right hand side in every iteration is given by $B^{-1}b$.
In this example for the the final tableau it holds that

$$B^{-1}b = \begin{bmatrix} \frac{3}{5} & -\frac{1}{5} \\ -\frac{1}{5} & \frac{2}{5} \end{bmatrix} \begin{bmatrix} b \\ 2b \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \end{bmatrix} \Leftrightarrow$$
$$\begin{bmatrix} b \\ 2b \end{bmatrix} = \begin{bmatrix} \frac{3}{5} & -\frac{1}{5} \\ -\frac{1}{5} & \frac{2}{5} \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 5 \\ 10 \end{bmatrix}$$

This means that $b = 5$

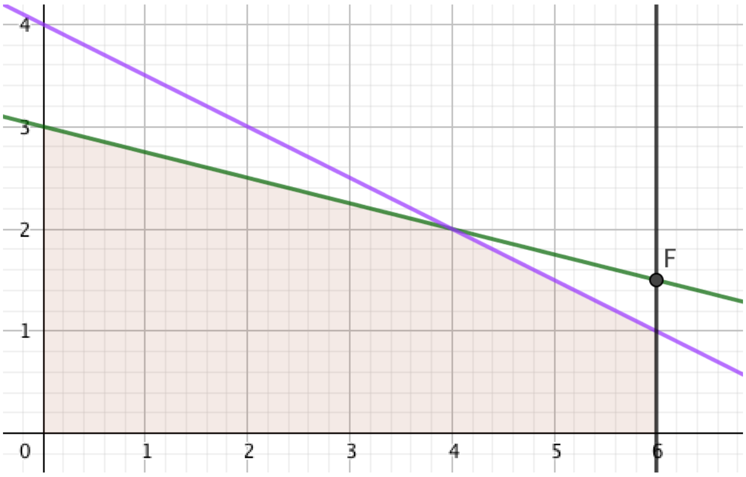
The optimal objective value is calculated in two different ways.
Using the coefficients: $Z = c_1x_1 + c_2x_2 + c_3x_3 = \frac{3}{2} \cdot 0 + 2 \cdot 1 + 3 \cdot 3 = 11$
Using fundamental insight and the value of b: $Z = c_b B^{-1}b = \begin{bmatrix} \frac{3}{5} & \frac{4}{5} \end{bmatrix} \begin{bmatrix} 5 \\ 10 \end{bmatrix} = 11$

4.4 Exercise 4

1. True: If the solutions is optimal but not a CPF solution there exists two optimal CPF solutions. On the line between the two optimal CPF solutions there are infinite many optimal solutions.

2. True: Same argument as the above point. All solutions between to optimal CPF solutions are optimal as well.

3. False: It is not guaranteed that the system of equations has a solution. Furthermore the solutions to n constraints might not fulfill all of the constraints:



In the example in the figure above there are two variables and three constraints. The solutions to two constraints (green and black line) is not in the feasible region and therefore the statement is false.

5 Linear Programming: Duality and Sensitivity analysis

5.1 Exercise 1

The dual problem is a minimization problem with variables y_1 and y_2 which are associated with each of the constraints in the primal problem. The right hand side of the primal problem becomes the coefficients in the objective function in the dual problem. Similar the coefficients in the objective function in the primal problem becomes the right hand side of the constraints in the dual problem. The coefficients on the left hand side of the primal problem are transposed to construct the dual problem. All of the constraints in the dual problem are "$\geq$"-constraints since the variables in the primal problem are ≥ 0. Similar the variables in the dual problem are ≥ 0 since the constraints in the primal problem are "$\leq$"-constraints.

$$\begin{aligned} \min \quad & Z = 12y_1 + y_2 && (51) \\ \text{s.t.} \quad & y_1 + y_2 \geq -1 && (52) \\ & y_1 + y_2 \geq -2 && (53) \\ & 2y_1 - y_2 \geq -1 && (54) \\ & y_1 \geq 0, y_2 \geq 0 && (55) \end{aligned}$$

5.2 Exercise 2

Using sensitivity analysis we want to determine the allowable range for c_1 and c_2. In other words we need to know the value of c_1 and c_2 such that the last basis remains unchanged.

From the tableau we read

$$S^*A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, S^* = \begin{bmatrix} 1 & -2 \\ 1 & -1 \end{bmatrix}, y^*A - c = \begin{bmatrix} 0 & 0 \end{bmatrix}, y^* = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

First we attempt changes to c_1. This gives $\Delta = \begin{bmatrix} \delta & 0 \end{bmatrix}$ and $\Delta_B = \begin{bmatrix} 0 & \delta \end{bmatrix}$ since $x_B = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$. From the slides we know that we need to ensure

$$y^*A - c + (\Delta_B S^*A - \Delta) \geq \mathbf{0} \quad \text{and} \quad y^* + \Delta_B S^* \geq \mathbf{0}$$

Typing in the values we get:

$$\begin{aligned} y^*A - c + (\Delta_B S^*A - \Delta) &= \begin{bmatrix} 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & \delta \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} \delta & 0 \end{bmatrix} \\ &= \begin{bmatrix} \delta & 0 \end{bmatrix} - \begin{bmatrix} \delta & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \geq \begin{bmatrix} 0 & 0 \end{bmatrix} \end{aligned}$$

and

$$y^* + \Delta_B S^* = \begin{bmatrix} 1 & 1 \end{bmatrix} + \begin{bmatrix} 0 & \delta \end{bmatrix} \begin{bmatrix} 1 & -2 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \end{bmatrix} + \begin{bmatrix} \delta & -\delta \end{bmatrix} \geq \begin{bmatrix} 0 & 0 \end{bmatrix}$$

From the last inequality we get $1 + \delta \geq 0 \Leftrightarrow \delta \geq -1$ and $1 - \delta \geq 0 \Leftrightarrow \delta \leq 1$ thus, $\delta \in [-1; 1]$ and $c_1 \in [2; 4]$ since c_1 originally was 3.

Next we attempt changes to c_2. This gives $\Delta = \begin{bmatrix} 0 & \delta \end{bmatrix}$ and $\Delta_B = \begin{bmatrix} \delta & 0 \end{bmatrix}$ since $x_B = \begin{bmatrix} x_2 \\ x_1 \end{bmatrix}$.

$$\begin{aligned} y^*A - c + (\Delta_B S^*A - \Delta) &= \begin{bmatrix} 0 & 0 \end{bmatrix} + \begin{bmatrix} \delta & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} 0 & \delta \end{bmatrix} \\ &= \begin{bmatrix} 0 & \delta \end{bmatrix} - \begin{bmatrix} 0 & \delta \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix} \geq \begin{bmatrix} 0 & 0 \end{bmatrix} \end{aligned}$$

and

$$y^* + \Delta_B S^* = \begin{bmatrix} 1 & 1 \end{bmatrix} + \begin{bmatrix} \delta & 0 \end{bmatrix} \begin{bmatrix} 1 & -2 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \end{bmatrix} + \begin{bmatrix} \delta & -2\delta \end{bmatrix} \geq \begin{bmatrix} 0 & 0 \end{bmatrix}$$

From the last inequality we get $1+\delta \geq 0 \Leftrightarrow \delta \geq -1$ and $1-2\delta \geq 0 \Leftrightarrow \delta \leq \frac{1}{2}$ thus, $\delta \in [-1; \frac{1}{2}]$ and $c_2 \in [-3; -\frac{3}{2}]$ since c_2 originally was -2.
The other part of the exercise consists of the determining the allowable range for the right hand side b_1 and b_2. For the final tabular to stay optimal we need to make sure that the right hand sides are greater than 0. When changing the right hand side we add Δ to respectively b_1 and b_2. Determining the allowable range for b_1 includes using fundamental insight: $S^*b + S\begin{bmatrix}\Delta \\ 0\end{bmatrix} \geq 0$
From the final simplex tabular we know $S^* = \begin{bmatrix}1 & -2 \\ 1 & -1\end{bmatrix}$ and $b = \begin{bmatrix}30 \\ 10\end{bmatrix}$
Calculating the allowable range for b_1:

$$S^*b + S\begin{bmatrix}\Delta \\ 0\end{bmatrix} = \begin{bmatrix}1 & -2 \\ 1 & -1\end{bmatrix}\begin{bmatrix}30 \\ 10\end{bmatrix} + \begin{bmatrix}1 & -2 \\ 1 & -1\end{bmatrix}\begin{bmatrix}\Delta \\ 0\end{bmatrix} = \begin{bmatrix}10 \\ 20\end{bmatrix} + \begin{bmatrix}\Delta \\ \Delta\end{bmatrix} = \begin{bmatrix}10+\Delta \\ 20+\Delta\end{bmatrix} \geq 0$$

The interval for Δ:

$$10 + \Delta \geq 0 \Rightarrow \Delta \geq -10$$
$$20 + \Delta \geq 0 \Rightarrow \Delta \geq -20$$

For Δ it holds that $\Delta \geq -10$. Adding this value to the original value of b_1 gives the allowable range for b_1

$$30 - 10 \leq b_1$$

For all values of b_1 greater than 20 the final tabular will remain optimal.
The same procedure is made for b_2.

$$S^*b + S\begin{bmatrix}0 \\ \Delta\end{bmatrix} = \begin{bmatrix}1 & -2 \\ 1 & -1\end{bmatrix}\begin{bmatrix}30 \\ 10\end{bmatrix} + \begin{bmatrix}1 & -2 \\ 1 & -1\end{bmatrix}\begin{bmatrix}0 \\ \Delta\end{bmatrix} = \begin{bmatrix}10 \\ 20\end{bmatrix} + \begin{bmatrix}-2\Delta \\ -\Delta\end{bmatrix} = \begin{bmatrix}10-2\Delta \\ 20-\Delta\end{bmatrix} \geq 0$$

The interval for Δ:

$$10 - 2\Delta \geq 0 \Rightarrow \Delta \leq 5$$
$$20 - \Delta \geq 0 \Rightarrow \Delta \leq 20$$

For Δ it holds that $\Delta \leq 5$. Adding this value to the original value of b_2 gives the allowable range for b_1

$$b_2 \leq 10 + 5$$

For all value of b_2 less than 15 the final tabular will remain optimal.

5.3 Exercise 3

Given the problem

$$\begin{aligned} \max \quad & Z = 5x_1 + 4x_2 \\ \text{s.t.} \quad & 2x_1 + 3x_2 \geq 10 \\ & x_1 + 2x_2 = 20 \\ & x_1 \in \mathbb{R}, x_2 \geq 0 \end{aligned}$$

a) Find the dual problem using the SOB method

First step in the SOB method is to construct the objective function for the dual problem. The coefficients for the dual problems objective function is found as the coefficients of the primal problems right hand side.
The objective function in the dual problem is Min $\quad 10y_1 + 20y_2$

Next step is to construct the constraints in the dual problem. Every variable in the primal problem gives a constraint in the dual problem, so we'll end up with 2 constraints in the dual problem.

- The coefficients to the constraints in the dual problem is found as the columns in the A matrix for the primal problem
- The right hand side of the constraints is the coefficients of the objective function in the primal problem
- The type of constraint ($\leq, \geq =$) is determined by the domain of the variable in the primal problem corresponding to the constraint. See the slides for the table.

This gives us the 2 constraints:

$$\begin{aligned}
&2y_1 + y_2 = 5 \\
&3y_1 + 2y_2 \geq 4
\end{aligned}$$

The third and last step in the SOB method is to determine the domain of the dual variables. The domain depends on the constraints type in the primal problem. A table of the conversions can be found in the slides. In our case the domain of the dual variables are $y_1 \leq 0$ and $y_2 \in \mathbb{R}$
Combining all the steps we get the final dual problem

$$\begin{aligned}
\min \qquad & W = 10y_1 + 20y_2 \\
\text{s.t.} \qquad & 2y_1 + y_2 = 5 \\
& 3y_1 + 2y_2 \geq 4 \\
& y_1 \leq 0, y_2 \in \mathbb{R}
\end{aligned}$$

b) Use Table 6.12 to convert the primal problem to our standard form given in the beginning of Sec. 6.1, and construct the corresponding dual problem. Then show that this dual problem is equivalent to the one obtained in part (a)

From table 6.12 we see that because x_1 is unconstrained, we need to introduce $x_1 = x_1^+ - x_1^-$

$$\begin{aligned}
\max \qquad & Z = 5(x_1^+ - x_1^-) + 4x_2 \\
\text{s.t.} \qquad & 2(x_1^+ - x_1^-) + 3x_2 \geq 10 \\
& (x_1^+ - x_1^-) + 2x_2 = 20 \\
& x_1^+ \geq 0, x_1^- \geq 0, x_2 \geq 0
\end{aligned}$$

Next we need to convert the $=$ constraint to a $\leq$ constraint. From table 6.12 the conversion is seen to be done by introducing two new constraints: $(x_1^+ - x_1^-) + 2x_2 \leq 20$ and $-(x_1^+ - x_1^-) - 2x_2 \leq -20$. So the primal problem on standard form is

$$\begin{aligned}
\max \qquad & Z = 5(x_1^+ - x_1^-) + 4x_2 \\
\text{s.t.} \qquad & 2(x_1^+ - x_1^-) + 3x_2 \geq 10 \\
& (x_1^+ - x_1^-) + 2x_2 \leq 20 \\
& -(x_1^+ - x_1^-) - 2x_2 \leq -20 \\
& x_1^+ \geq 0, x_1^- \geq 0, x_2 \geq 0
\end{aligned}$$

At last we need to change the $\geq$ constraint to a $\leq$ constraint, which simply is done by multiplying by -1

$$\begin{aligned} \max \quad & Z = 5(x_1^+ - x_1^-) + 4x_2 \\ \text{s.t.} \quad & -2(x_1^+ - x_1^-) - 3x_2 \leq -10 \\ & (x_1^+ - x_1^-) + 2x_2 \leq 20 \\ & -(x_1^+ - x_1^-) - 2x_2 \leq -20 \\ & x_1^+ \geq 0, x_1^- \geq 0, x_2 \geq 0 \end{aligned}$$

Now that the problem is on standard form we can easily construct the dual problem.

$$\begin{aligned} \min \quad & W = -10y_1 + 20y_2 - 20y_3 \\ \text{s.t.} \quad & -2y_1 + y_2 - y_3 \geq 5 \\ & 2y_1 - y_2 + y_3 \geq -5 \\ & -3y_1 + 2y_2 - 2y_3 \geq 4 \\ & y_1 \geq 0, y_2 \geq 0, y_3 \geq 0 \end{aligned}$$

If we now introduce $y_1' = -y_1$ and $y_2 - y_3 = y_2'$ we get

$$\begin{aligned} \min \quad & W = 10y_1 + 20y_2 \\ \text{s.t.} \quad & 2y_1 + y_2 = 5 \\ & 3y_1 + 2y_2 \geq 4 \\ & y_1 \leq 0, y_2 \in \mathbb{R} \end{aligned}$$

which is exactly the same we got by using the SOB method.

5.4 Exercise 4

From the optimal table we can see that $y^* = \begin{bmatrix} 1 & 1 \end{bmatrix}$. By knowing the production costs of the new product we can construct the corresponding column in the A matrix, i.e $A_e = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$. From earlier we know that the optimality test is given by $y^* A_e - c_e$ and if it less than zero the current solution is not the optimal solution.

So if we want to know what the profit should be before the new product enters basis, i.e becomes part of the optimal solution, we need to solve the inequality $y^* A_e - c_e < 0$, where c_e is the profit.

$y^* A_e - c_e = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} - c_e < 0 \Leftrightarrow 5 - c_e < 0 \Leftrightarrow c_e > 5$

The profit of the new product should be greater than 5.

5.5 Exercise 5

Replacing b to $\bar{b}$ will only change the objective function in the dual problem, so the domain of solutions will remain the same. y^* is a legal solution to the dual problem and $c\overline{x}$ is the optimal solution to the new primal problem. From the weak duality theorem this gives us that $c\overline{x} \leq y^* b$.

6 Week 6: Modeling with Integer variables

6.1 Exercise 1

Introduce a binary variable x_i that is equal to 1 if one invest in project i and 0 if one do not invest in project i. The IP model is then

$$\begin{aligned} \max \quad & Z = x_1 + 1.8x_2 + 1.6x_3 + 0.8x_4 + 1.4x_5 && (56) \\ \text{s.t.} \quad & 6x_1 + 12x_2 + 10x_3 + 4x_4 + 8x_5 \leq 20 && (57) \\ & x_i \in \{0, 1\} && (58) \end{aligned}$$

The following piece of Julia code solves the problem

```
using JuMP
using GLPKMathProgInterface
# Notice that we have to use GLPKSolverMIP and not GLPKSolverLP
# in order to solve integer programs.
m = Model(solver = GLPKSolverMIP())
# here we define 5 variables, all are binary
@variable(m, x[i=1:5], Bin)
@objective(m, Max, x[1] + 1.8x[2] + 1.6x[3]+ 0.8x[4] + 1.4x[5])
@constraint(m, 6x[1] + 12x[2] + 10x[3] + 4x[4] + 8x[5] <= 20 )
status = solve(m)
println("Objective value: ", getobjectivevalue(m))
println("value of x variables: ", getvalue(x))
```

The optimal solution to the problem is
$Z = 3.4$
$x_1 = x_3 = x_4 = 1$
$x_2 = x_5 = 0$

6.2 Exercise 2

x_i is the production level of product i. Introduce a binary variable y_i that indicates whether a product is being produced or not. We have to ensure that x_i only has a value when y_i is equal to 1. This can be done with the following constraint:

$$x_i \leq My_i \qquad \forall i = 1, 2, 3, 4 \tag{59}$$

where M_i is a large number greater or equal to the maximum number of products that can be produced. From the third constraint it can be seen that no more than 6000 products can be produced. Furthermore it can be seen that no more than $6000/3 = 2000$ products can be produced as the smallest coefficient on the left hand side is 3. The constraint, ensuring that x_i only has a value when y_i is equal to 1, can be formulated as

$$x_i \leq 2000y_i \qquad \forall i = 1, 2, 3, 4 \tag{60}$$

Now we have to model the three described constraints

1. No more than two of the products can be produced: $y_1 + y_2 + y_3 + y_4 \leq 2$
2. Either product 3 or 4 can be produced only if either product 1 or 2 is produced. First we model that product 3 only can be produced if product 1 or product 2 (or both) are produced:

   $$y_1 + y_2 \geq y_3 \tag{61}$$

   The same holds for product 4

   $$y_1 + y_2 \geq y_4 \tag{62}$$

3. How to model this kind of constraint is described in "Introduction to operations research" by Hillier and Lieberman on page 484.
   We introduce a binary variable u.

$$5x_1 + 3x_2 + 6x_3 + 4x_4 \leq 6000 + Mu \tag{63}$$
$$4x_1 + 6x_2 + 3x_3 + 5x_4 \leq 6000 + M(1-u) \tag{64}$$
$$\tag{65}$$

   For M it should hold that $M + 6000$ should be greater than the left hand side. We already know that $x_i \leq 2000$ which means that the left hand side in both constraint is at most $(5+3+6+4) \cdot 2000 = 36000$. This means that we can chosse $M = 30000$

The problem can be solved with Julia as:

```
using GLPKMathProgInterface

m = Model(solver = GLPKSolverMIP())
@variable(m,x[1:4]>=0)
@variable(m,y[1:4],Bin)
@variable(m,u,Bin)

M=30000

@objective(m, Max, 70*x[1]+60*x[2]+90*x[3]+80*x[4]
     50000*y[1] 40000*y[2] 70000*y[3] 60000*y[4])

@constraint(m, [i=1:4] ,x[i]<=2000*y[i])
@constraint(m,y[1]+y[2]+y[3]+y[4]<=2)
@constraint(m,y[1]+y[2]>=y[3])
@constraint(m,y[1]+y[2]>=y[4])
@constraint(m,5*x[1]+3*x[2]+6*x[3]+5*x[4]<=6000+M*u)
@constraint(m,4*x[1]+6*x[2]+3*x[3]+5*x[4]<=6000+M*(1 u))

status = solve(m)

println("Objective value: ",getobjectivevalue(m))
for i=1:4
    println("x", i , "=" , getvalue(x[i]))
end

for i=1:4
    println("y", i , "=" , getvalue(y[i]))
end
println("u=" , getvalue(u))
```

The objective value is Z=80000, $x_2 = 2000$, $y_2 = 1$ and $u = 0$. All other variables are equal to 0

6.3 Exercise 3

We introduce the following integer variables:

- x_{11}, $x_{21} \geq 0$: Pieces of toy one and two produced at factory one.
- x_{12}, $x_{22} \geq 0$: Pieces of toy one and two produced at factory two.

In order to help us define the objective function and constraints we introduce the following binary variables:

- $y_1 \in \{0, 1\}$: 1 if factory one is used else 0
- $y_2 \in \{0, 1\}$: 1 if factory two is used else 0
- $t_1 \in \{0, 1\}$: 1 if toy one is produced else 0
- $t_2 \in \{0, 1\}$: 1 if toy two is produced else 0

With the introduced variables the objective function is:

$$\max \quad 10\,(x_{11} + x_{12}) + 15\,(x_{21} + x_{22}) - 50000t_1 - 80000t_2$$

Just one factory would be used corresponds to the constraint:

$$y_1 + y_2 \leq 1$$

Toy 1 can be produced at the rate of 50 per hour in factory 1 and 40 per hour in factory 2. Toy 2 can be produced at the rate of 40 per hour in factory 1 and 25 per hour in factory 2. Factories 1 and 2, respectively, have 500 hours and 700 hours of production time available before Christmas that could be used to produce these toys. This can be formulated as the following two constraints:

$$\frac{1}{50}x_{11} + \frac{1}{40}x_{21} \leq 500y_1$$

$$\frac{1}{40}x_{12} + \frac{1}{25}x_{22} \leq 700y_2$$

To ensure that toy one is only produced if we decide to produce toy one and likewise with toy two we need to add the following two constraints:

$$x_{11} + x_{12} \leq M_1 t_1$$

$$x_{21} + x_{22} \leq M_2 t_2$$

At last we need to find some good values for M_1 and M_2. Some good values could be $M_1 = \max\{500 \cdot 50, 700 \cdot 40\}$ and $M_2 = \max\{500 \cdot 40, 700 \cdot 25\}$ as these are the maximum production capacity of toy 1 and toy 2.

6.3.1 Julia code

```
using JuMP
using GLPKMathProgInterface

m = Model(solver = GLPKSolverMIP())

@variable(m, x[1:2,1:2] >= 0)
@variable(m, y[1:2], Bin)
@variable(m, z[1:2], Bin)


@objective(m, Max, 10*(x[1,1]+x[1,2]) + 15*(x[2,1]+x[2,2])   50000*y[1]   80000*y[2])

@constraint(m, z[1] + z[2] <= 1)
@constraint(m, (1/50)*x[1,1] + (1/40)*x[2,1] <= 500*z[1])
@constraint(m, (1/40)*x[1,2] + (1/25)*x[2,2] <= 700*z[2])
@constraint(m, x[1,1] + x[1,2] <= 28000*y[1])
@constraint(m, x[2,1] + x[2,2] <= 20000*y[2])

status = solve(m)

println("Objective value: ", getobjectivevalue(m))
```

```
println("x11 = ", getvalue(x[1,1]))
println("x12 = ", getvalue(x[1,2]))
println("x21 = ", getvalue(x[2,1]))
println("x22 = ", getvalue(x[2,2]))
println("y1 = ", getvalue(y[1]))
println("y2 = ", getvalue(y[2]))
println("z1 = ", getvalue(z[1]))
println("z2 = ", getvalue(z[2]))
```

6.4 Solution to exercise 4

There are several ways to formulate the MIP. One of them is the following

$$\max x_1 + 2x_2 + 3x_3$$

subject to

$$x_1 - x_2 = 3y \tag{66}$$
$$x_1 + 3x_2 + 5x_3 \leq 10.7 \tag{67}$$
$$-2 \leq y \leq 2 \tag{68}$$
$$x_1, x_2, x_3 \geq 0 \tag{69}$$
$$y \in \mathbb{Z} \tag{70}$$

To see that this is equivalent to the model in the exercise, let's first consider the non-linear constraint

$$|x_1 - x_2| = 0 \text{ or } 3 \text{ or } 6$$

from the model in the exericse. This constraint is identical to the constraint

$$x_1 - x_2 = -6 \text{ or } -3 \text{ or } 0 \text{ or } 3 \text{ or } 6$$

In the solution we have that y should be an integer between -2 and 2. This means that $3y$ can take one of the five values -6,-3,0,3 or 6 and that means that constraint (66) together with the definition of y does exactly the same as the non-linear constraint from the exercise.
The model can be implemented in Julia as shown below. The optimal solution is $x_1 = 7.175$, $x_2 = 1.175$, $x_3 = 0$ and $y = 2$. The optimal objective value is 9.525.

```
using JuMP
using GLPKMathProgInterface
m = Model(solver = GLPKSolverMIP())
@variable(m, x1 >= 0)
@variable(m, x2 >= 0)
@variable(m, x3 >= 0)
@variable(m, -2 <= y <= 2, Int)
@objective(m, Max, x1  + 2x2 + 3x3 )
@constraint(m, x1 - x2 -3y == 0 )
@constraint(m, x1 + 3x2 + 5x3 <= 10.7 )
solve(m)
println("Objective value: ", getobjectivevalue(m))
println("x1 = ", getvalue(x1))
println("x2 = ", getvalue(x2))
println("x3 = ", getvalue(x3))
println("y = ", getvalue(y))
```

6.5 Solution to exercise 5

We define six binary variables y_{1j} and y_{2j} for $j = 1, 2, 3$ with y_{ij} having the following interpretation

$$y_{ij} = \begin{cases} 1 & \text{if } x_i = j \\ 0 & \text{otherwise} \end{cases} \tag{71}$$

We just need three binary variables per x variable since each x-variable only can take the values 0,1,2 and 3 because of the constraint in the original model. The value $x_i = 0$ is obtained by setting $y_{ij} = 0$ for $j = 1, 2, 3$ so we do not need a y_{i0} variable.
With these binary variable we can rewrite the model using only y_{ij}. Everywhere we have x_i we can substitute with $1y_{i1}+2y_{i2}+3y_{i3}$, everywhere we have x_i^2 we can substitute with $1^2y_{i1}+2^2y_{i2}+3^2y_{i3}$, everywhere we have x_i^3 we can substitute with $1^3y_{i1}+2^3y_{i2}+3^3y_{i3}$ and everywhere we have x_i^4 we can substitute with $1^4y_{i1}+2^4y_{i2}+3^4y_{i3}$. With this we get the objective and constraint (72) of the model below. Constraints (73) and (74) are added to ensure that at most one of the variables y_{i1}, y_{i2} and y_{i3} take a non-zero value, $i = 1, 2$ (see definition (71)).

$$\begin{aligned} \max Z =& 4\left(1^2y_{11} + 2^2y_{12} + 3^2y_{13}\right) - \left(1^3y_{11} + 2^3y_{12} + 3^3y_{13}\right) \\ &+ 10\left(1^2y_{21} + 2^2y_{22} + 3^2y_{23}\right) - \left(1^4y_{21} + 2^4y_{22} + 3^4y_{23}\right) \end{aligned}$$

subject to

$$1y_{11} + 2y_{12} + 3y_{13} + 1y_{21} + 2y_{22} + 3y_{23} \le 3 \tag{72}$$

$$y_{11} + y_{12} + y_{13} \le 1 \tag{73}$$

$$y_{21} + y_{22} + y_{23} \le 1 \tag{74}$$

$$y_{11}, y_{12}, y_{13}, y_{21}, y_{22}, y_{23} \in \{0, 1\} \tag{75}$$

One way of implementing the model in Julia is show below. The model makes heavy use of the sum(... for ...) function. We could also have written the model in a more explicit way (typing in all the coefficients directly). The sum notation makes it easier to extend and/or change the model.

```
using JuMP
using GLPKMathProgInterface
m = Model(solver = GLPKSolverMIP())
@variable(m, y[1:2,1:3], Bin)
@objective(m, Max,
  4(sum(j^2 * y[1,j] for j=1:3))
- sum(j^3 * y[1,j] for j=1:3)
+ 10(sum(j^2 * y[2,j] for j=1:3))
- sum(j^4 * y[2,j] for j=1:3))
@constraint(m, sum(j * y[1,j] for j=1:3) + sum(j * y[2,j] for j=1:3) <= 3 )
@constraint(m, [i=1:2], sum(y[i,j] for j=1:3) <= 1)
print(m)
status = solve(m)
println("Objective value: ", getobjectivevalue(m))
yVal = getvalue(y)
println("y = ", yVal)
println("x1 (computed) = ", sum(j*yVal[1,j] for j=1:3))
println("x2 (computed) = ", sum(j*yVal[2,j] for j=1:3))
```

7 Week 7: Modeling with Integer variables II

7.1 Exercise 1

As the leader of an oil-exploration drilling venture, you must determine the least-cost selection of five out of ten possible sites. Label the sites $S_1, S_2, S_3, \ldots, S_{10}$ and the exploration costs associated with each as $C_1, C_2, \ldots, C_{10}$. Regional development restrictions are such that:

- Evaluating sites S_1, and S_7 will prevent you from exploring site S_8
- Evaluating site S_3 or S_4 prevents you from assessing site S_5
- Of the group S_3, S_6, S_7, and S_8, only two sites may be assessed

Formulate an integer program to determine the minimum-cost exploration scheme that satisfies these restrictions

$$x_j = \begin{cases} 1 \text{ if site } S_j \text{ is selected } j = 1, 2, 3, \ldots, 10 \\ 0 \text{ otherwise} \end{cases}$$

$$\begin{aligned} \texttt{Minimize:} \quad & \sum_{j=1}^{10} C_j \cdot x_j \\ \texttt{s.t.} \quad x_1 + x_7 + x_8 &\leq 2 \\ x_3 + x_5 &\leq 1 \\ x_4 + x_5 &\leq 1 \\ x_3 + x_6 + x_7 + x_8 &\leq 2 \\ \sum_{j=1}^{n} x_j &= 5 \\ x_j &\in \{0, 1\} \qquad \text{for } i = 1, 2, \ldots, 10 \end{aligned}$$

7.2 Exercise 2

Suppose that a mathematical model fits linear programming except for the restrictions that

- At least one of the following two inequalities holds

$$\begin{aligned} 3x_1 - x_2 - x_3 + x_4 &\leq 12 \\ x_1 + x_2 + x_3 + x_4 &\leq 15 \end{aligned}$$

- At least two of the following three inequalities holds

$$\begin{aligned} 2x_1 + 5x_2 - x_3 + x_4 &\leq 30 \\ -x_1 + 3x_2 + 5x_3 + x_4 &\leq 40 \\ 3x_1 - x_2 + 3x_3 - x_4 &\leq 60 \end{aligned}$$

Show how to reformulate these restrictions to fit an MIP model.

- At least one of the following two inequalities holds. Introduce one binary variable y which provides the possibility to activate/deactivate the constraints

$$\begin{aligned} 3x_1 - x_2 - x_3 + x_4 &\leq 12 + M \cdot (1 - y) \\ x_1 + x_2 + x_3 + x_4 &\leq 15 + M \cdot y \end{aligned}$$

- At least two of the following three inequalities holds. Introduce three binary variables y_1, y_2, and y_3 which provide the possibility to deactivate the constraints.

$$
\begin{aligned}
2x_1 + 5x_2 - x_3 + x_4 &\leq 30 + M \cdot y_1 \\
-x_1 + 3x_2 + 5x_3 + x_4 &\leq 40 + M \cdot y_2 \\
3x_1 - x_2 + 3x_3 - x_4 &\leq 60 + M \cdot y_3 \\
\sum_{i=1}^{3} y_i &\leq 1
\end{aligned}
$$

7.3 Exercise 3

Consider the following integer program:

$$
\begin{aligned}
\texttt{Maximize } Z &= x_1 + 5x_2 \\
x_1 + 10x_2 &\leq 20 \\
x_1 &\leq 2 \\
x_1, x_2 &\in \mathbb{Z}_+
\end{aligned}
$$

1. Use a binary representation of the variables to reformulate this model as a Binary Integer Program

   We can deduce from the formulations that the upperbounds of variables x_1 and x_2 are both 2. This implies that N equals 1 in both cases. To be able to represent all integer values for both x_1 and x_2 we therefore need two binary variables for each. We introduce y_0 and y_1 for x_3 and y_2 and y_3 for x_3.

$$
\begin{aligned}
x_1 &= y_0 + 2y_1 \\
x_2 &= y_2 + 2y_3
\end{aligned}
$$

$$
\begin{aligned}
\texttt{Maximize:} \quad & y_0 + 2y_1 + 5y_2 + 10y_3 \\
\texttt{s.t} \quad y_0 + 2y_1 + 10y_2 + 20y_3 &\leq 20 \\
y_0 + 2y_1 &\leq 2 \\
y_0 + 2y_1 &\geq 1 \\
y_2 + 2y_3 &\geq 1 \\
y_i &\in \{0, 1\} \quad \text{for } i = 1, 2, 3, 4
\end{aligned}
$$

2. Solve the model using a computer and use the optimal solution to identify an optimal solution to the original problem

$$y_1 = y_2 = 1, \text{ with optimal objective value: } 7. \text{ The solution implies } x_1 = 2 \text{ and } x_2 = 1$$

```
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

@variable(m, y[1:4], Bin)

@objective(m, Max, y[1]+2*y[2]+5*y[3]+10*y[4])

@constraint(m, y[1]+2*y[2]+10*y[3]+20*y[4]<=20)
@constraint(m, y[1]+2*y[2] <= 2)
@constraint(m, y[1]+2*y[2] >= 1)
@constraint(m, y[3]+2*y[4] >= 1)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of y variables: ", getvalue(y))
```

Figure 3: Julia implementation

7.4 Exercise 4

The Fly-Right Airplane Company builds small jet planes to sell to corporations for the use of the executives. To meet the needs of these executives, the company's customers sometimes order a custom design of the air planes being purchased. When this occurs, a substantial start up cost is incurred to intiate production of these airplanes.

Fly-Right has recently received purchase requests from three customers with short deadlines. However, as the company's production facilities already are almost completely tied up filling previous orders, it will not be able to accept all three orders. Therefore a decision now needs to be made on the number of airplanes the company will agree to produce (if any) for each of the three customers.

The table below gives the relevant information on each of the three customers. Each customer has a fixed start-up cost, paid just once irrespective of how many planes are produced for that customer. The marginal net revenue is received per plane produced. The third row gives the percentage of available capacity needed for each aircraft.The last row states the number of airplanes ordered by each customer (less will obviously be accepted).

	Customer		
	1	2	3
Start-up cost	\$3 million	\$ 2 million	0
Marginal net revenue	\$ 2 million	\$ 3 million	\$ 0.8 million
Capacity used per plane	20 %	40 %	20%
Maximum order	3 planes	2 planes	5 planes

1. Formulate an integer linear programming model that can be used to maximize Fly-Right's total profit (net revenue - start up costs)

$$y_i = \begin{cases} 1 \text{ if customer } i \text{ is selected } i = 1, 2 \\ 0 \text{ otherwise} \end{cases}$$

$$x_i = \text{Number of planes produced for customer } i = 1, 2, 3$$

$$
\begin{aligned}
\texttt{Maximize:} \quad & 2x_1 + 3x_2 + 0.8x_3 - 3y_1 - 2y_2 \\
\texttt{s.t} \quad & 0.2x_1 + 0.4x_2 + 0.2x_3 \leq 1.0 \\
& x_1 \leq 3 \cdot y_1 \\
& x_2 \leq 2 \cdot y_2 \\
& x_3 \leq 5 \\
& x_i \geq 0 \quad \text{for } i = 1,2,3 \\
& y_i \in \{0,1\} \quad \text{for } i = 1,2
\end{aligned}
$$

2. Solve your model using a computer and state the number of airplanes produced for each customer

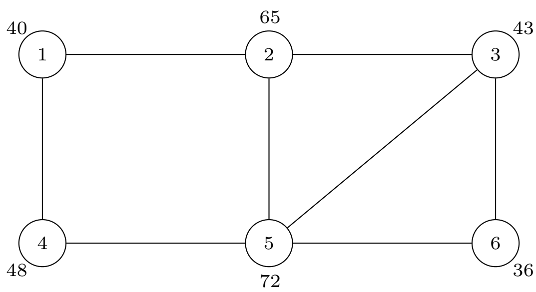
   The optimal solution is $x_1 = 0, x_2 = 2, x_3 = 1, y_1 = 0$ and $y_2 = 1$, and the optimal objective value is 4.8 million dollars. The solutions states that two airplanes should be produced for Customer 2 and one for Customer 3.

```julia
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

@variable(m, x[1:3] >= 0)
@variable(m, y[1:2], Bin)


@objective(m, Max,  2*x[1]+3*x[2]+0.8*x[3]-3*y[1]-2*y[2])

@constraint(m, x[1]<=3*y[1])
@constraint(m, x[2]<=2*y[2])
@constraint(m, x[3]<=5)
@constraint(m, 0.2*x[1]+0.4*x[2]+0.2*x[3]<=1)


solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
println("Value of y variables: ", getvalue(y))
```

Figure 4: Julia implementation

7.5 Exercise 5

The following map shows six intersections at which automatic traffic monitoring devices might be installed. A station at any particular node can monitor all the road links meeting that intersection. Numbers next to nodes reflect the monthly cost (in thousands of dollars) of operating a station at that location.



1. Formulate the problem of providing full coverage at minimum total cost as a set covering problem

$$x_i = \begin{cases} 1 \text{ if a station is installed at node } i = 1,2,3,4,5,6 \\ 0 \text{ otherwise} \end{cases}$$

$$\begin{aligned} \texttt{Minimize:} \quad & 40x_1 + 65x_2 + 43x_3 + 48x_4 + 72x_5 + 36x_6 \\ \texttt{s.t} \quad & x_1 + x_2 \geq 1 \\ & x_1 + x_4 \geq 1 \\ & x_2 + x_5 \geq 1 \\ & x_4 + x_5 \geq 1 \\ & x_2 + x_3 \geq 1 \\ & x_3 + x_5 \geq 1 \\ & x_3 + x_6 \geq 1 \\ & x_5 + x_6 \geq 1 \\ & y_i \in \{0,1\} \qquad \text{for } i = 1,2,3,4,5,6 \end{aligned}$$

2. Find the optimal solution to your model using a computer

   The optimal solution is $x_1 = x_3 = x_5 = 1, x_2 = x_4 = x_6 = 0$, and the optimal objective value is 155 thousand dollars. The solution states that stations should be opened at nodes one, three, and five.

```julia
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

@variable(m, x[1:6], Bin)

cost = [40 65 43 48 72 36]

type Arc
    from::Int64
    to::Int64
end

arcs = [Arc(1,4), Arc(2,5), Arc(4,5), Arc(2,3), Arc(1,2), Arc(3,5), Arc(3,6), Arc(5,6)]

@objective(m, Min, sum(cost[i]*x[i] for i=1:6))

for arc in arcs
    @constraint(m, x[arc.from]+x[arc.to]>=1)
end

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
```

Figure 5: Julia implementation

3. Revise your formulation in Part 1) to obtain a binary integer programming that minimizes the number of uncovered road links while using at most two stations

$$x_i = \begin{cases} 1 \text{ if a station is installed at node } i = 1,2,3,4,5,6 \\ 0 \text{ otherwise} \end{cases}$$

$$y_j = \begin{cases} 1 \text{ if a road link } j \text{ is left uncovered for } j = 1,2,3,4,5,6,7,8 \\ 0 \text{ otherwise} \end{cases}$$

$$
\begin{aligned}
\texttt{Minimize:}\quad & y_1 + y_2 + y_3 + y_4 + y_4 + y_5 + y_6 + y_7 + y_8 \\
\texttt{s.t}\quad & x_1 + x_2 + y_1 = 1 \\
& x_1 + x_4 + y_2 = 1 \\
& x_2 + x_5 + y_3 = 1 \\
& x_4 + x_5 + y_4 = 1 \\
& x_2 + x_3 + y_5 = 1 \\
& x_3 + x_5 + y_6 = 1 \\
& x_3 + x_6 + y_7 = 1 \\
& x_5 + x_6 + y_8 = 1 \\
& \sum_{i=1}^{6} x_i = 2 \\
& x_i \in \{0,1\} \qquad \text{for } i = 1,2,3,4,5,6 \\
& y_i \in \{0,1\} \qquad \text{for } i = 1,2,3,4,5,6,7,8
\end{aligned}
$$

4. Find the optimal solution to the revised problem using a computer

   The optimal solution is $x_1 = x_5 = 1, x_2 = x_3 = x_4 = x_6 = 0$, and the optimal objective value is two uncovered roads. The solution states that stations should be opened at nodes one and five.

```
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

@variable(m, x[1:6], Bin)
@variable(m, y[1:8], Bin)

type Arc
    from::Int64
    to::Int64
end

arcs = [Arc(1,4), Arc(2,5), Arc(4,5), Arc(2,3), Arc(1,2), Arc(3,5), Arc(3,6), Arc(5,6)]

@objective(m, Min, sum(y[i] for i=1:8))

for i=1:length(arcs)
    arc = arcs[i]
    @constraint(m, x[arc.from]+x[arc.to]+y[i]>=1)
end

@constraint(m, sum(x[i] for i=1:6)==2)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
println("Value of y variables: ", getvalue(y))
```

Figure 6: Julia implementation

7.6 Exercise 6

Air Anton is a small commuter airline running six flights per day from New York City to surrounding resort areas. Flight crews are all based in New York, working flights to various locations and then returning on the next flight home. Taking into account complex work rules and pay incentives Air Anton schedulers have constructed the 8 possible work patterns detailed in the following table. Each row of the table indicates the flights that are covered in a particular pattern ($\times$) and the daily cost per crew (in thousands of dollars).

The company want to choose the minimum total cost collection of work patterns that cover all flights exactly once.

	Flight						
Pattern	1	2	3	4	5	6	Cost
1	-	×	-	×	-	-	1.40
2	×	-	-	-	-	×	0.96
3	-	×	-	×	×	-	1.52
4	-	×	-	-	×	×	1.60
5	×	-	×	-	-	×	1.32
6	-	-	×	-	×	-	1.12
7	-	-	-	×	-	×	0.84
8	×	-	×	×	-	-	1.54

1. Formulate the problem of providing full coverage at minimum total cost as a set partitioning problem

$$x_i = \begin{cases} 1 \text{ if pattern } i \text{ is assigned to a flight crew for } i = 1, 2, 3, 4, 5, 7, 8 \\ 0 \text{ otherwise} \end{cases}$$

$$\begin{aligned} \texttt{Minimize:} \quad & \sum_{i=1}^{6} c_i x_i \\ \texttt{s.t} \quad & \sum_{i=1}^{6} a_{if} x_i = 1 \quad \forall f = 1, 2, 3, \ldots, 6 \\ & x_i \in \{0, 1\} \quad \text{for } i = 1, 2, \ldots, 8, \end{aligned}$$

where c_i states the cost of work pattern i, a_{fi} is a binary parameter indicating whether or not flight $f = 1, 2, \ldots, 6$ is contained in work pattern $i = 1, 2, \ldots, 8$

Written in full this model looks like

$$\begin{aligned} \texttt{Minimize:} \quad & 1.4x_1 + 0.96x_2 + 1.52x_3 + 1.60x_4 + 1.32x_5 + 1.12x_6 + 0.84x_7 + 1.54x_8 \\ \texttt{s.t} \quad & x_2 + x_5 + x_8 = 1 \\ & x_1 + x_3 + x_4 = 1 \\ & x_5 + x_6 + x_8 = 1 \\ & x_1 + x_3 + x_7 + x_8 = 1 \\ & x_3 + x_4 + x_6 = 1 \\ & x_2 + x_4 + x_5 + x_7 = 1 \\ & x_i \in \{0, 1\} \quad \text{for } i = 1, 2, \ldots, 8. \end{aligned}$$

2. Find the optimal solution to your model using a computer

   The optimal solution is $x_3 = x_5 = 1, x_1 = x_2 = x_4 = x_6 = x_7 = x_8 = 0$, and the optimal objective value is \$2840. The solution states that work patterns three and five cover the six flights at minimum cost.

```
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

n=8

@variable(m, x[1:n], Bin)

cost = [1.40 0.96 1.52 1.60 1.32 1.12 0.84 1.54]

A = [0.0 1.0 0.0 0.0 1.0 0.0 0.0 1.0;
    1.0 0.0 1.0 1.0 0.0 0.0 0.0 0.0;
    0.0 0.0 0.0 0.0 1.0 1.0 0.0 1.0;
    1.0 0.0 1.0 0.0 0.0 0.0 1.0 1.0;
    0.0 0.0 1.0 1.0 0.0 1.0 0.0 0.0;
    0.0 1.0 0.0 1.0 1.0 0.0 1.0 0.0]

@objective(m, Min, sum(cost[i]*x[i] for i=1:n))

@constraint(m, [j=1:size(A)[1]], sum(A[j,i]*x[i] for i=1:n) == 1)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
```

Figure 7: Julia implementation

8 Week 8: Modeling with Integer variables III

8.1 Exercise 1

You are the marketing manager of a large company and are looking at a number of possible advertising campaigns in order to attract more customers. Six campaigns are possible, and they are detailed in the table below. Each campaign requires a certain investment (in millions of dollars) and will yield a certain number of new customers (in the thousands). At most 5 million dollars is available for the campaigns.

Campaign	Investment	Return
Superbowl half-time Adv.	3M	80
Radio Adv. Campaign	800K	20
Television (Non peak hour)	500K	22
City Newspaper	2M	75
Viral Marketing Campaign	50K	4
Web advertising	600K	10

1. Assuming it is possible to invest in fractions of a campaign, but not more than one of each, formulate a Linear Program that can be used to maximize the number of new customers.

$$x_i = \text{Fraction of campaign } i = 1, 2, \ldots, 6 \text{ chosen}$$

$$\texttt{Maximize:} \quad \sum_{i=1}^{6} p_i x_i$$

$$\texttt{s.t.} \sum_{i=1}^{6} w_i x_i \leq 5$$

$$0 \leq x_i \leq 1 \qquad \text{for } i = 1, 2, \ldots, 6,$$

where w_i and p_i denote the investment required the return from campaign i, respectively.

2. Solve your model with a computer.

```
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())
n=6

profit = [80 20 22 75 4 10]
weight = [3 0.8 0.5 2 0.05 0.6]

@variable(m, 0 <= x[1:n] <= 1 )
@objective(m, Max, sum(profit[i]*x[i] for i=1:n))
@constraint(m, sum(weight[i]*x[i] for i=1:n) <= 5)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
```

Figure 8: Julia implementation

The optimal solution is $x_1 = 0.816667, x_3 = x_4 = x_5 = 1, x_2 = 0$, and the optimal objective value is 166.33 thousand customers. The solution states that campaigns three, four, and five cover will be selected completely, while campaign one will also be chosen, but with value 0.816667.

3. Solve the problem again, this time using a greedy algorithm in which at each iteration you increase, as much as possible the value of the variable x_j that maximizes the ratio $\frac{p_i}{w_i}$, where p_i denotes the profit of investment campaign i and w_i refers to the investment required for campaign i. What do you observe? Can you prove that greedy is an optimal strategy in this case?

   Ordering the campaigns in decreasing order of this ratio yields

	Campaigns					
	5	3	4	1	2	6
$\frac{p}{w}$	80	44	36	26.67	25	16.67

Table 4: Ratio Order

   (a) Campaign 5 has the largest ratio, investment required is less than available budget, therefore we first set $x_5 = 1$, remaining budget = 5.0-0.05 = 4.95

   (b) Campaign 3 has the 2nd best ratio, investment required is less than available budget, therefore we then set $x_3 = 1$, remaining budget = 4.95-0.50 = 4.45

   (c) Campaign 4 has the 3rd best ratio, investment required is less than available budget, therefore we then set $x_4 = 1$, remaining budget = 4.45-2.0 = 2.45

   (d) Campaign 1 has the 4th best ratio, the investment required exceeds our available budget, select as much of x_4 as we can, i.e, $x_4 = \frac{2.45}{3} = 0.816667$

   We have exhausted the available budget, this solution is identical to that found in Part 2).

   Proof

   Assume we are given a set $I = \{1, 2, \ldots, n\}$ of items. Suppose further that item j gives the best return for the amount invested (i.e, $\frac{p_j}{w_j} \geq \frac{p_i}{w_i}$ $\forall k \in I$ with $i \neq j$). Let us assume that the optimal solution does not contain item j. Then, there must exist an item k such that $\frac{p_k}{w_k} < \frac{p_j}{w_j}$. If we replace an ϵ amount of item k with an ϵ amount of item j, the objective function improves by $\epsilon \cdot (\frac{p_j}{w_j} - \frac{p_k}{w_k})$. Therefore, the optimal solution must contain item j at its greatest value. This is precisely what the greedy algorithm ensures.

4. Assume now that you must invest in a campaign in its entirety or not at all. Update your model from part 1) to reflect this.

$$x_i = \begin{cases} 1 \text{ if campaign } i = 1, 2, \ldots, 6 \text{ is chosen} \\ 0 \text{ otherwise} \end{cases}$$

$$\texttt{Maximize:} \quad \sum_{i=1}^{6} p_i x_i$$

$$\texttt{s.t.} \sum_{i=1}^{6} w_i x_i \leq 5$$

$$x_i \in \{0, 1\} \quad \text{for } i = 1, 2, \ldots, 6,$$

   where w_i and p_i denote the investment required the return from campaign i, respectively.

5. Solve your model with a computer

```
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())
n=6

profit = [80 20 22 75 4 10]
weight = [3 0.8 0.5 2 0.05 0.6]

@variable(m, x[1:n], Bin )
@objective(m, Max, sum(profit[i]*x[i] for i=1:n))
@constraint(m, sum(weight[i]*x[i] for i=1:n) <= 5)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))

```

Figure 9: Julia implementation

The optimal solution is $x_1 = x_4 = 1, x_2 = x_3 = x_5 = 0, x_6 = 0$, and the optimal objective value is 155 thousand customers . The solution states that only campaigns one and four are chosen.

6. Solve this binary integer program again using the same greedy algorithm above. What do you observe?

   (a) Campaign 5 has the largest ratio, investment required is less than available budget, therefore we first set $x_5 = 1$, remaining budget = 5.0-0.05 = 4.95
   (b) Campaign 3 has the 2nd best ratio, investment required is less than available budget, therefore we then set $x_3 = 1$, remaining budget = 4.95-0.50 = 4.45
   (c) Campaign 4 has the 3rd best ratio, investment required is less than available budget, therefore we then set $x_4 = 1$, remaining budget = 4.45-2.0 = 2.45
   (d) Campaign 1 has the 4th best ratio, the investment required exceeds our available budget, set $x_1 = 0$, remaining budget = 2.45
   (e) Campaign 2 has the 5th best ratio; investment required is less than available budget, therefore we then set $x_2 = 1$, remaining budget = 2.45-0.8 =1.65
   (f) Campaign 6 has the worst ratio; investment required is less than available budget, therefore we then set $x_6 = 1$, remaining budget = 1.75-0.6 =1.05

   This solution states that we should run all campaigns except the first, and yields 131 thousand new customers. The greedy approach failed to find the optimal solution.

7. You have been given a revised analysis of the marketing campaigns with the following information

Campaign	Investment	Return
Superbowl half-time Adv.	1M	80
Radio Adv. Campaign	1.8M	20
Television (Non peak hour)	1.5M	22
City Newspaper	1.1M	75
Viral Marketing Campaign	2.2M	4
Web advertising	2M	10

   Resolve your binary integer program with the computer and using your greedy algorithm. What do you observe?

   Re-ordering the campaigns in decreasing order of the $\frac{p}{w}$ ratio yields

   Applying the greedy algorithm yields

   (a) Campaign 1 has the largest ratio, investment required is less than available budget, therefore we first set $x_5 = 1$, remaining budget = 5.0-1.0 = 4.0

	Campaigns					
	1	4	3	2	6	5
$\frac{p}{w}$	80	68.18	14.67	11.11	5.0	1.818

Table 5: Ratio Order

(b) Campaign 4 has the 2nd best ratio, investment required is less than available budget, therefore we then set $x_4 = 1$, remaining budget = 4.0-1.1 = 2.9

(c) Campaign 3 has the 3rd best ratio, investment required is less than available budget, therefore we then set $x_3 = 1$, remaining budget = 2.9-1.5 = 1.4

(d) All other campaigns now require an investment that exceeds the available budget, therefore we set $x_2 = x_6 = x_5 = 0$.

This solution states that we should run investment campaigns one, three, and four, giving 177 thousand new customers.

Solving the mathematical model from Part 4 above with the updated parameters yields the same solution.

```julia
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())
n=6

profit = [80 20 22 75 4 10]
weight = [1 1.8 1.5 1.1 2.2 2]

@variable(m, x[1:n], Bin )
@objective(m, Max, sum(profit[i]*x[i] for i=1:n))
@constraint(m, sum(weight[i]*x[i] for i=1:n) <= 5)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
```

Figure 10: Julia implementation

8. You suspect the greedy approach might be optimal for 0-1 knapsack problems with this structure. Identify the structure, and then prove that the greedy approach will always yield the optimal solution.

   Ordering the items by decreasing return yields the same sequence as ordering the items by increasing investment cost.

   Proof

   Similar exchange argument to Part 4.

8.2 Exercise 2

Consider the following binary integer program.

$$
\begin{aligned}
\texttt{Minimize}\ & 15x_1 + 18x_2 + 6x_3 + 20x_4 \\
\texttt{s.t.}\ & x_1 + x_4 \geq 1 \\
& x_1 + x_2 + x_4 \geq 1 \\
& x_2 + x_3 + x_4 \geq 1 \\
& x_i \in \{0,1\} \quad \text{for } i = 1,2,3,4
\end{aligned}
$$

You would like to solve this problem using a greedy algorithm.

1. Explain why it seems reasonable to choose a free x_j to fix at the value one by picking the variable with the least ratio

$$r_j = \frac{\text{cost coefficient of variable } j}{\text{number of uncovered elements that variable } j \text{ covers}}$$

   The proposed ratio explicitly seeks minimum cost by including the objective coefficient in its numerator. Still, it also considers feasibility by dividing by the number of still uncovered rows or elements each free j could resolve. The effect is to seek the most efficient next choice x_j to fix=1, the best in the myopic sense.

2. Determine the solution obtained by the greedy algorithm

   Initial ratios are given in Table 6.

	Variables			
	1	2	3	4
r	7.50	9.00	6.00	6.67

Table 6: Initial Ratios

   (a) Variable three has the best ratio, set $x_3 = 1$, updated ratios: $r_1 = 7.5, r_2 = 18, r_4 = 10$

   (b) Variable one has the best ratio, set $x_1 = 1$, all rows now covered. Stop. Solution has a value of 21

3. Solve the problem using a computer and comment on the result

   By inspection we can see that the optimal has a value of 20, and has $x_4 = 1$

```julia
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

@variable(m, x[1:4], Bin)
@objective(m, Min, 15*x[1]+18*x[2]+6*x[3]+20*x[4])

@constraint(m, x[1]+x[4]>=1)
@constraint(m, x[1]+x[2]+x[4]>=1)
@constraint(m, x[2]+x[3]+x[4]>=1)

solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
```

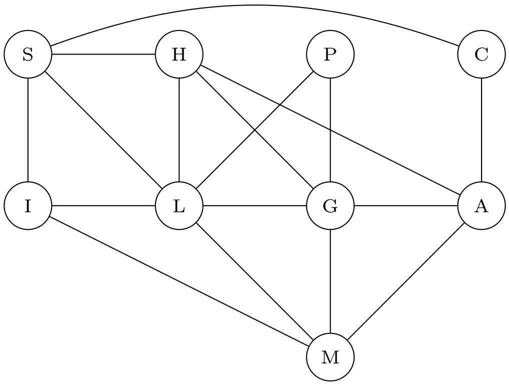
Figure 11: Julia implementation

8.3 Exercise 3

Suppose that you are responsible for scheduling times for lectures in a university. You want to make sure that any two lectures with a common student occur at different times to avoid a conflict. We could put the various lectures on a chart and mark with an $\times$ any pair that has students in common:

A more convenient representation of this information is a graph with one vertex for each lecture and in which two vertices are joined if there is a conflict between them

Lecture	A	C	G	H	I	L	M	P	S
(A)stronomy		×	×	×			×		
(C)hemistry	×							×	
(G)reek	×			×		×	×	×	
(H)istory	×		×			×			×
(I)talian						×	×		×
(L)atin			×	×	×		×	×	×
(M)usic	×		×		×	×			
(P)hilosphy			×			×			
(S)panish		×		×	×	×			



Now, we cannot schedule two lectures at the same time if there is a conflict, but we would like to use as few separate times as possible, subject to this constraint. How many different times are necessary? We can code each time with a colour, for example 11:00-12:00 might be given the colour green, and those lectures that meet at this time will be coloured green. The no-conflict rule then means that we need to colour the vertices of our graph in such away that no two adjacent vertices (representing courses which conflict with each other) have the same colour.

One can apply the following greedy algorithm to colour the graph

1. colour a vertex with colour 1
2. Pick an uncoloured vertex v. colour with the lowest-numbered colour the has not been used on any previously coloured adjacent vertices v. If all previously-used colours appear on vertices adjacent to v, then we must introduce a new colour and number it.
3. Repeat the previous step until all vertices are coloured.

Now, answer the following questions:

1. Using the set of colours {Green=1, Red=2, Blue=3, Yellow=4, and Cyan = 5}, colour the vertices in the order G, L, H, P, M, A, I, S, C using the greedy algorithm above. How many colours do you need?

   Four colours are needed, see the graph

2. Using the set of colours {Green=1, Red=2, Blue=3, Yellow=4, and Cyan = 5}, colour the vertices in the order A, I, P, M, S, C, H, L,G using the greedy algorithm above. How many colours do you need? Comment on the results to parts 1) and 2).

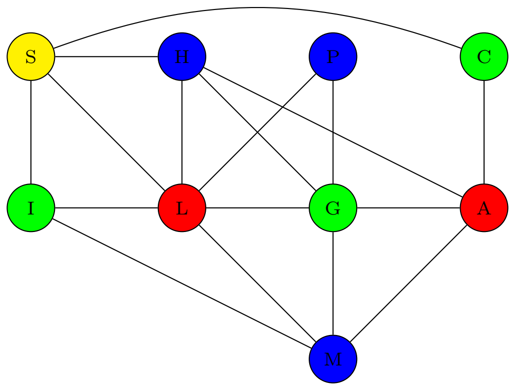


Figure 12: Question one colouring with four colours

Five colours are needed, see the graph

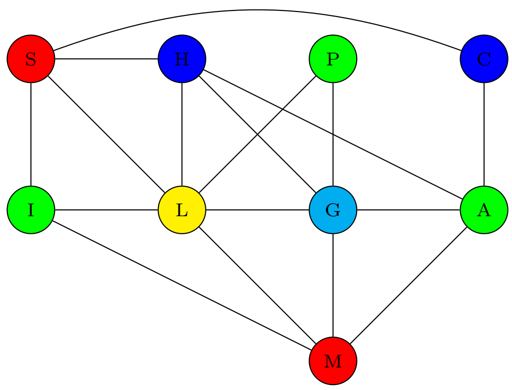


Figure 13: Question two colouring with five colours

The greedy approach is clearly not optimal. It provides solutions of different quality depending on which node is used to initialize the algorithm.

3. Formulate the graph colouring problem as an integer programming problem

   We number the nodes as follows S=1, H=2, P=3, C=4, A=6, G=6, L=7, I=8, and M=9.

$$x_{ij} = \begin{cases} 1 \text{ if node } i \text{ receives colour } j \text{ for } i = 1, 2, \ldots, 9 \text{ and } j = 1, 2, \ldots 4 \\ 0 \text{ otherwise} \end{cases}$$

$$y_j = \begin{cases} 1 \text{ if colour } j \text{ is used for } j = 1, 2, 3, 4 \\ 0 \text{ otherwise} \end{cases}$$

$$\texttt{Minimize:} \quad \sum_{j=1}^{4} y_j$$

$$\sum_{j=1}^{4} x_{ij} = 1 \qquad \text{for } i = 1, 2, \ldots, 9$$

$$\sum_{i=1}^{9} x_{ij} \leq 9 \cdot y_j \qquad \text{for } j = 1, 2, 3, 4$$

$$x_{ij} + x_{i',j} \leq 1 \qquad \text{for all edges } (i, i'), j = 1, 2, 3, 4$$

$$x_{ij} \in \{0, 1\} \quad i = 1, 2, \ldots, 9, \; j = 1, 2, 3, 4$$

$$y_j \in \{0, 1\} \quad j = 1, 2, 3, 4$$

4. Solve your model using a computer

   Four colours are needed in the optimal solution.

   The optimal solution is $x_{14} = x_{21} = x_{31} = x_{41} = x_{53} = x_{62} = x_{73} = x_{81} = x_{94} = 1$ and $y_1 = y_2 = y_3 = y_4 = 1$. All other variables have a value of zero. The found colouring is visualized below.

```julia
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

nc = 4
nv = 9

@variable(m, x[1:nv,1:nc], Bin)
@variable(m, y[1:nc], Bin)

type Arc
    from::Int64
    to::Int64
end

arcs = [Arc(1,2), Arc(1,7), Arc(1,8), Arc(1,4), Arc(2,5), Arc(2,6), Arc(2,7),
Arc(3,6), Arc(3,7), Arc(4,5), Arc(5,6), Arc(5,9), Arc(6,7), Arc(6,9), Arc(7,8),
Arc(7,9), Arc(8,9)]

@objective(m, Min, sum(y[i] for i=1:nc))

@constraint(m,[i=1:nv], sum(x[i,j] for j=1:nc)==1)
@constraint(m,[j=1:nc], sum(x[i,j] for i=1:nv)<=nv*y[j])

for i=1:length(arcs)
    for j=1:nc
        arc = arcs[i]
        @constraint(m, x[arc.from,j]+x[arc.to,j]<=1)
    end
end


solve(m)

println("Objective Value: ", getobjectivevalue(m))
println("Value of x variables: ", getvalue(x))
println("Value of y variables: ", getvalue(y))
```

Figure 14: Julia implementation

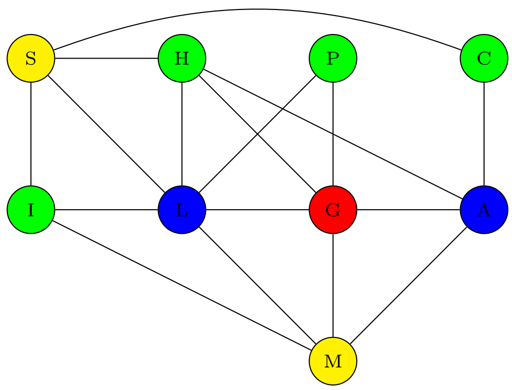


Figure 15: Colouring obtained by solving the mathematical model

9 Week 9: Modeling with Integer variables IV

9.1 Exercise 1

Recall the marketing campaign problem from Week 8, Part 4), in which you had decide how many advertising campaigns (from a set of six possible) you would run, given a maximum budget of 5 million dollars, in order to maximize the number of new customers. Each campaign required a certain investment (in millions of dollars) and would yield a certain number of new customers (in the thousands). You were given the following data.

Campaign	Investment	Return
Superbowl half-time Adv.	3	80
Radio Adv. Campaign	0.80	20
Television (Non peak hour)	0.50	22
City Newspaper	2	75
Viral Marketing Campaign	0.05	4
Web advertising	0.6	10

You have applied a greedy heuristic and obtained a solution in which you will invest in all campaigns except the Superbowl half-time advertisement. This strategy would lead to 131,000 new customers. Based on the model formulation given in ... the solution is $x_1 = 0$ and $x_2, x_3, x_4, x_5, x_6 = 1$

1. You decide to try and improve this solution by considering the neighbourhood of solutions obtained by removing one used campaign and inserting an unused campaign. How many feasible neighbouring solutions are there?

   (a) $x_1 = 1, x_2 = 0 \Rightarrow$ infeasible

   (b) $x_1 = 1, x_3 = 0 \Rightarrow$ infeasible

   (c) $x_1 = 1, x_4 = 0 \Rightarrow$ feasible

   (d) $x_1 = 1, x_5 = 0 \Rightarrow$ infeasible

   (e) $x_1 = 1, x_6 = 0 \Rightarrow$ infeasible

   There is only one feasible solution. We can swap the values of x_1 and x_4.

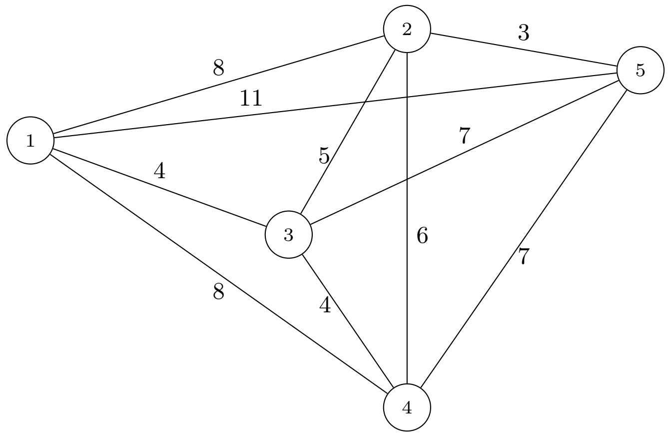
2. Given this neighbourhood, is there an improving move?

   Swapping the values of x_1 and x_4, leads to an improvement of 5,000 customers.

3. Update your solution. Is it locally optimal? Explain your answer. The new solution is $x_4 = 0$ and $x_1, x_2, x_3, x_5, x_6 = 1$. This is locally optimally given the specified neighbourhood. No neighbouring solution has a better objective.

9.2 Exercise 2

Consider the Travelling Salesman Problem shown below, where City 1 is considered to be the home city.



1. How many possible tours are there (excluding tours that are simply the reverse of others)?

   For a symmetric TSP with n cities and a specified starting city there are $\frac{n-1!}{2}$. Therefore there a $\frac{4!}{2} = 12$

2. Starting at City 1, apply the nearest neighbour heuristic to obtain a feasible solution. What is its objective value?

   The resulting tour is $1 - 3 - 4 - 2 - 5 - 1$. This has an objective value of 28.

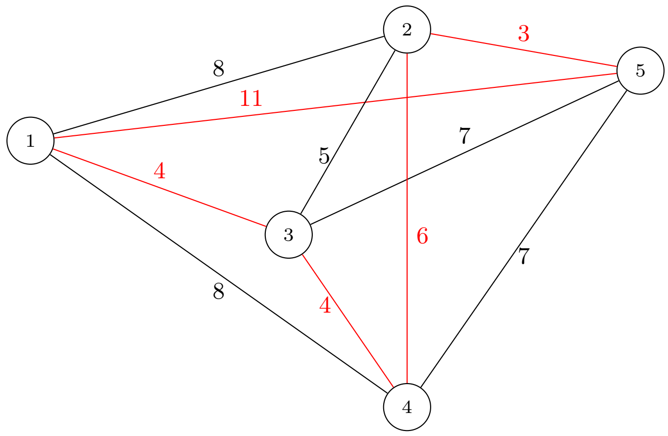


Figure 16: Nearest neighbour solution

3. Consider all possible 2-opt edge swaps. How many lead to different feasible tours?

   There are five edge pair swaps that lead to feasible tours.

4. Given the solution $1 - 3 - 4 - 2 - 5 - 1$, perform a 2-opt swap on edges $(2, 4)$ and $(1, 5)$. What is the new tour, and what is its objective value?

   The resulting tour is $1 - 3 - 4 - 5 - 2 - 1$. This has an objective of 26.

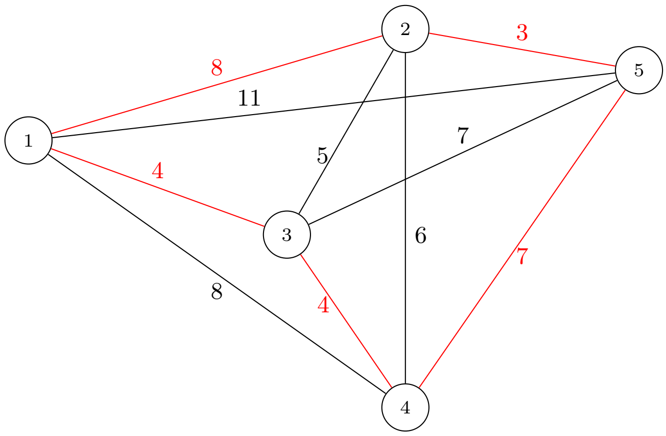


Figure 17: Updated solution

5. Find the optimal solution using Julia. You might like to make use of the Julia code given in Lecture 9.

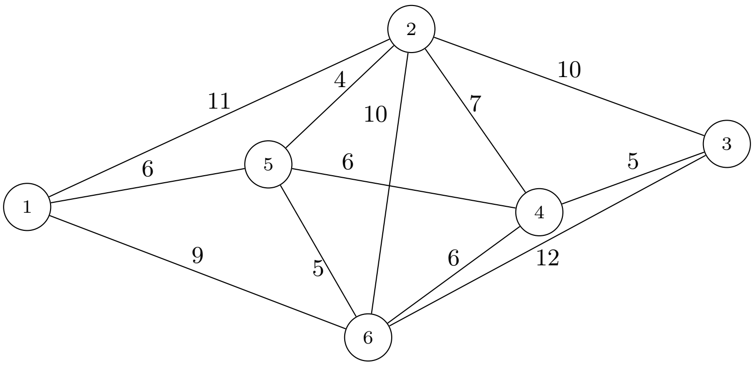
   The optimal solution is $x_{12} = x_{25} = x_{31} = x_{43} = x_{54} = 1$ and has an objective value of 26.

```julia
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())
n=5
cost = [0 8 4 8 11; 8 0 5 6 3; 4 5 0 4 7; 8 6 4 0 7; 11 3 7 7 0]

#variable an objective definition
@variable(m, x[1:n,1:n], Bin)
@variable(m, u[1:n])
@objective(m, Min, sum(cost[i, j] * x[i, j] for i = 1:n for j = 1:n))

#constraints specifying a node is not connected to itself
@constraint(m, notself[i=1:n], x[i,i] == 0)
#constraints specifying one edge leaves each node
@constraint(m, oneout[i=1:n], sum(x[i,j=1:n]) == 1)
#constraints specifying one edge enters each node
@constraint(m, onein[j=1:n], sum(x[i=1:n,j]) == 1)
#position in tour contraints
for i=1:n, j=1:n
    if i != 1 && j!=1
        @constraint(m, u[i]-u[j]+(n-1)*x[i,j] <= (n-2))
    end
end
#u-variable values
@constraint(m, u[1]==1)
@constraint(m, oneposl[i=2:n], u[i]>=2)
@constraint(m, oneposu[i=2:n], u[i]<=5)
#solve the model
solve(m)

println("objective: ", getobjectivevalue(m))
for i=1:n, j=1:n
   if getvalue(x[i,j]) > 1-1e-6
      println("Edge ", i, "-", j, " ", getvalue(x[i,j]))
   end
end
```

Figure 18: Julia implementation

9.3 Exercise 3

Consider the Traveling Salesman Problem shown below, where City 1 is the home city.



1. Starting at City 1, apply the nearest neighbour heuristic to obtain a feasible solution. What is its objective value?

   The resulting solution is $1-5-2-4-3-6-1$. This has an objective value of 43.

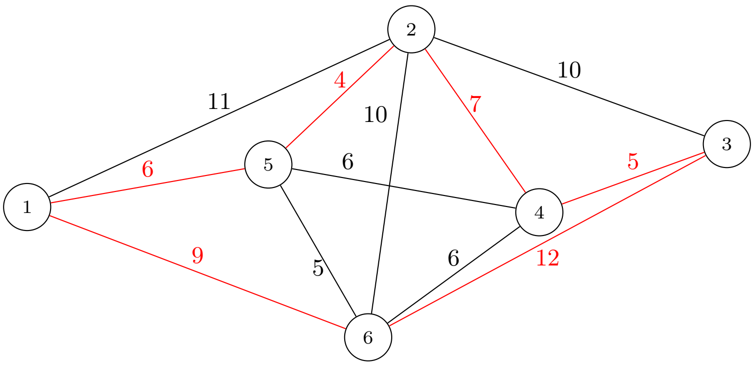


Figure 19: Nearest neighbour solution

2. Depending on where you start, why might the nearest neighbour heuristic fail to generate a feasible solution in this case?

   The graph is not complete. It is not possible to reach certain cities from others. For example, try running the nearest neighbour heuristic from City 2.

3. Consider a 3-opt exchange on edges $(2,4)$, $(1,6)$ and $(3,6)$. How many feasible new tours does this generate, and what are their respective objective values?

   There is only one feasible new tour, and this has an objective of 40.

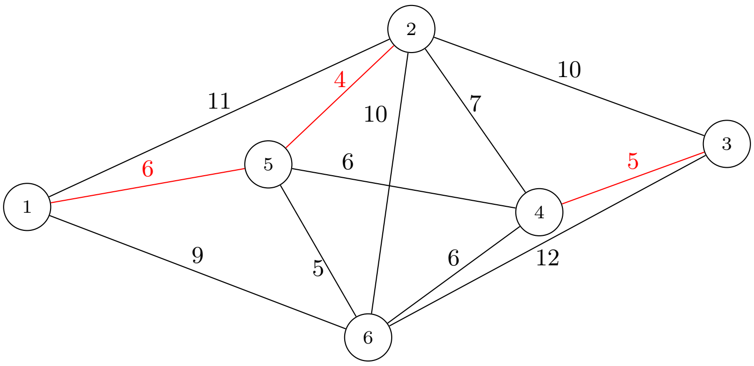


Figure 20: Edges removed

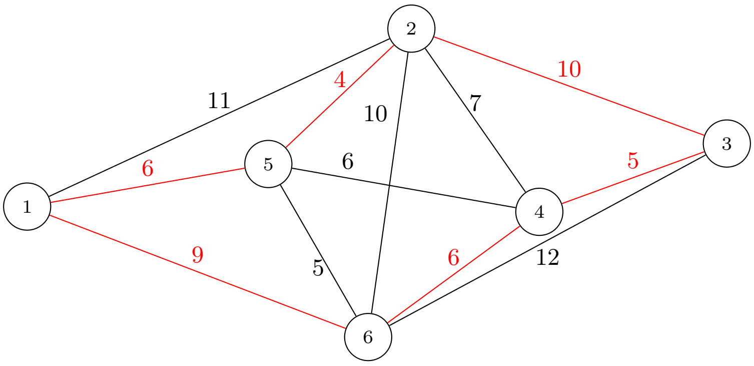


Figure 21: The only other possibility

4. Find the optimal solution using Julia. You might like to make use of the Julia code given in Lecture 9. (**Hint:** introduce highly penalized edges to generate a complete graph). How does this compare to your best found tour?

   The optimal solution is $x_{15} = x_{23} = x_{34} = x_{46} = x_{52} = x_{61} = 1$. This has an objective value of 40, and corresponds to the tour $1-5-2-3-4-6-1$. This is the same as the tour found by the 3-opt exchange.

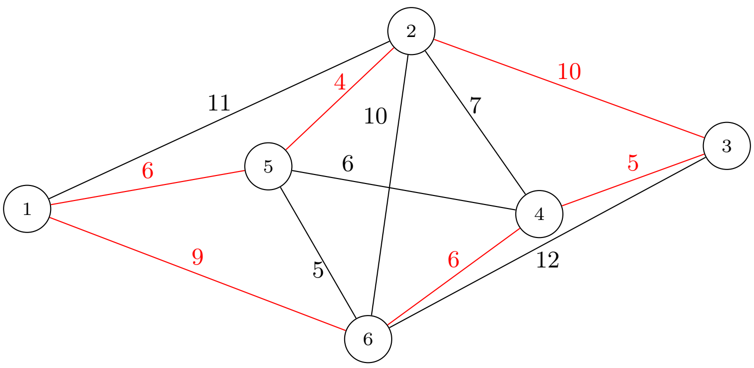


Figure 22: The optimal solution

```
using JuMP
using GLPKMathProgInterface

m=Model(solver=GLPKSolverMIP())

n=6

cost  =  [0 11 50 50 6 9; 11 0 10 7 4 10; 50 10 0 5 50 12; 50 7 5 0 6 6;
          6 4 50 6 0 5; 9 10 12 6 5 0]

#variable definition
@variable(m, x[1:n,1:n], Bin)
@variable(m, u[1:n])
#objective function
@objective(m, Min, sum(cost[i, j] * x[i, j] for i = 1:n for j = 1:n))
# constraints - x_{ii}=0
@constraint(m, notself[i=1:n], x[i,i] == 0)
# constraints - flow out of node
@constraint(m, oneout[i=1:n], sum(x[i,j=1:n]) == 1)
# constraints - flow into node
@constraint(m, onein[j=1:n], sum(x[i=1:n,j]) == 1)
# constraints - node order
for i=1:n, j=1:n
    if i != 1 && j!=1
        @constraint(m, u[i]-u[j]+(n-1)*x[i,j] <= (n-2))
    end
end
# constraints - bounds on u
@constraint(m, u[1]==1)
@constraint(m, oneposl[i=2:n], u[i]>=2)
@constraint(m, oneposu[i=2:n], u[i]<=6)
solve(m)
println("objective: ", getobjectivevalue(m))
for i=1:n, j=1:n
   if getvalue(x[i,j]) > 1-1e-6
      println("Edge ", i, "-", j, " ", getvalue(x[i,j]))
      #println("x:", )
   end
end
```

Figure 23: Julia implementation

10 Week 10, Totally Unimodular Matrices

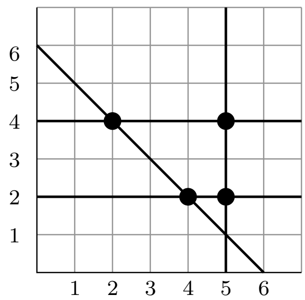
10.1 Exercise 1

Consider the integer program:

$$\text{Min} \quad 2.5x_1 + 7.9x_2 \tag{76}$$
$$\text{s.t.} \quad x_1 \leq 5 \tag{77}$$
$$x_2 \leq 4 \tag{78}$$
$$x_2 \geq 2 \tag{79}$$
$$x_1 + x_2 \geq 6 \tag{80}$$
$$x_1, x_2 \quad \text{integer} \tag{81}$$

(a) Does the feasible region have integer corner points? Hint: draw it on a graph.

Yes it does



(b) Is the constraint coefficient matrix totally unimodular? Hint: recall that for a 2×2 matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$, the determinant is $ad - bc$. The constraint matrix is

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

and all the square submatrices and their determinants are:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \longrightarrow \det = (1) \times (1) - 0 \times 0 = 1$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \longrightarrow \det = (1) \times (1) - 0 \times 0 = 1$$

$$\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \longrightarrow \det = (1) \times (1) - 0 \times (1) = 1$$

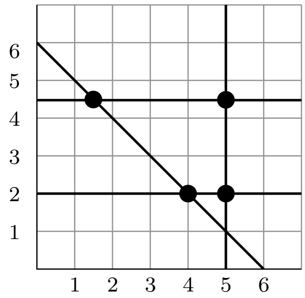
$$\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \longrightarrow \det = 0 \times (1) - (1) \times 0 = 0$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \longrightarrow \det = 0 \times (1) - (1) \times (1) = -1$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \longrightarrow \det = 0 \times (1) - (1) \times (1) = -1$$

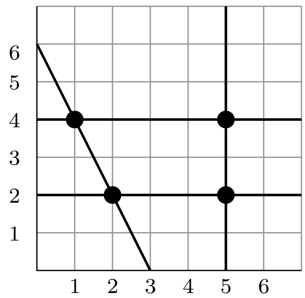
They are all either 0,1 or -1, so the matrix is totally unimodular.

(c) Repeat questions (a) and (b), but with the right-hand-side of constraint (3) changed to 4.5.



Now there are some corner points that are not integer, but nothing changed in the coefficient matrix, so it is still totally unimodular. This demonstrates that both total unimodularity and integer right hand side values are necessary for integral extreme points.

(d) Repeat questions (a) and (b), but with the coefficient of x_1 in constraint (5) changed to 2.



All the extreme points are still integer with this change. However, the constraint matrix is now

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 2 & 1 \end{bmatrix}$$

and the square submatrix

$$\begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix} \longrightarrow \det = 0 \times (1) - (1) \times (2) = -2$$

has a determinant of -2, so it is not totally unimodular. This shows that total unimodularity is only a sufficient condition, and not at all necessary for the feasible region to have integral extreme points.

(e) What if the objective function coefficients are changed?

The definition of total unimodularity is completely independent of the objective function coefficients, so it does not matter if they change, or even if they are not integer.

10.2 Exercise 2

Assume we have an assignment problem with three workers and three tasks. Each worker has to be assigned to exactly one task, and each task needs to be done by exactly one worker (any worker can be assigned to any task). Use the binary variable x_{ij} to denote whether or not worker i is assigned to task j, and write down the constraints.
The constraints are

$$
\begin{array}{ccccccccccccccccccl}
x_{11} & + & x_{12} & + & x_{13} & & & & & & & & & & & & & = 1 \\
& & & & & & x_{21} & + & x_{22} & + & x_{23} & & & & & & & = 1 \\
& & & & & & & & & & & & x_{31} & + & x_{32} & + & x_{33} & = 1 \\
x_{11} & & & + & & & x_{21} & & & + & & & x_{31} & & & & & = 1 \\
& & x_{12} & & & + & & & x_{22} & & & + & & & x_{32} & & & = 1 \\
& & & & x_{13} & & & + & & & x_{23} & & & + & & & x_{33} & = 1
\end{array}
$$

(a) Is the constraint coefficient matrix totally unimodular? **Hint: use slide 12 from the lecture.**

The constraint coefficient matrix is

$$
\begin{array}{cl}
\left[\begin{array}{ccccccccc}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
\hline
1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1
\end{array}\right]
&
\begin{array}{l}
\\ M_1(\text{workers}) \\ \\ \\ M_2(\text{tasks}) \\ \\
\end{array}
\end{array}
$$

and the rows have been partitioned into two sets. It easy to see for each column j that

$$\sum_{i \in M_1} a_{ij} - \sum_{i \in M_2} a_{ij} = 0.$$

(c) Would the optimal solution to the LP relaxation be integral?

Yes, the A matrix is totally unimodular. This implies that the extreme points of the feasible region are integral (our right hand side values are all 1.

(b) Repeat questions (a) and (b), but now assume that some workers can do multiple tasks and some tasks require multiple workers.

This only changes the right-hand side, and not the coefficients. So the matrix stays totally unimodular. Once again, as long as the right-hand side values are integral, the total unimodularity of the matrix implies that the extreme points of the polyhedron are integral, and thus the LP relaxation solution will be integral.

10.3 Exercise 3

Are the following matrices totally unimodular or not:

1.

$$
A_1 = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \end{bmatrix}
$$

No. Contains the following square submatrix, which has a determinant of 2

$$
\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}
$$

2.
$$A_2 = \begin{bmatrix} -1 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 \end{bmatrix}$$

Yes, we can partition the rows into two sets M_1 and M_2, where the first two rows belong to set M_1 and the last three to M_2

11 Branch & Bound

11.1 Exercise 1

The correct answers are:

- True
- True
- False

11.2 Exercise 2

The Binary Integer Program is formulated as:

$$\begin{aligned}
\max Z = 2x_1 - x_2 + 5x_3 - 3x_4 + 4x_5 \\
\text{s.t} \\
3x_1 - 2x_2 + 7x_3 - 5x_4 + 4x_5 \leq 6 \\
x_1 - x_2 + 2x_3 - 4x_4 + 2x_5 \leq 0 \\
x_1, x_2, x_3, x_4, x_5 \in \{0, 1\}
\end{aligned}$$

By solving the LP relaxation with `Julia` we find the solution

$$(x_1, x_2, x_3, x_4, x_5) = \left(\frac{2}{3}, 1, 1, 1, 1\right)$$

with the objective value $z_{LP} = 6.333$. Current best BIP solution $z^* = -\infty$.

Branch $x_1 = 0$: We add the constraint $x_1 = 0$ and solve the LP relaxation with `Julia` resulting in the solution

$$(x_1, x_2, x_3, x_4, x_5) = (0, 0, 1, 1, 1)$$

with the objective value $z_{LP} = 6$. Since it is a integer solution we update the current best BIP solution to $z^* = 6$.

Branch $x_1 = 1$: We add the constraint $x_1 = 1$ and solve the LP relaxation with `Julia` resulting in the solution

$$(x_1, x_2, x_3, x_4, x_5) = (1, 1, 0.8571, 1, 1)$$

with the objective value $z_{LP} = 6.286$. Normally we should go further down this branch since the solution is not an integer solution but because $\lfloor z_{LP} \rfloor \leq z^*$ we know that we wont find a better solution than the current solution $z^* = 6$.

The optimal solution to the BIP is $x_3 = x_4 = x_5 = 1$ with an objective value at $z^* = 6$.

11.3 Exercise 3

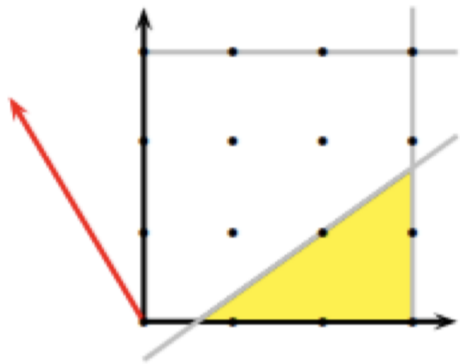
Solution to parts a) and b)

The MIP problem is:

$$\begin{aligned} \max \quad & -3x_1 + 5x_2 \\ \text{s.t.} \quad & \\ 5x_1 - 7x_2 \quad & \geq 3 \\ x_1 \quad & \leq 3 \\ x_2 \quad & \leq 3 \\ x_1, x_2 \quad & \geq 0 \end{aligned}$$

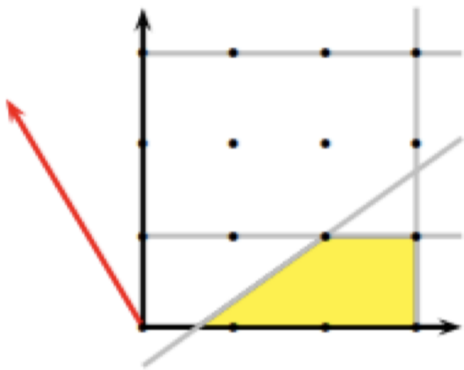
Where x_1 and x_2 are integers.

This can be illustrated graphically as



Here it can be easy seen that the optimal LP solution is $(x_1, x_2) = \left(3, \frac{12}{7}\right)$ which corresponds to an objective value at $z_{LP} = -\frac{3}{7}$. Current best MIP solution $z^* = -\infty$.

- **Branch:** $x_2 \leq 1$
  Now the problem can be illustrated graphically as



  It can be easy seen that the optimal LP solution is $(x_1, x_2) = (2, 1)$ which corresponds to an objective value at $z_{LP} = -1$. Current best MIP solution $z^* = -1$.

- **Branch:** $x_2 \geq 2$
  No solutions

The optimal solution to the MIP is $(x_1, x_2) = (2, 1)$ with an objective value at $z^* = -1$.

Solution to parts c) and d)
Let $x_1 = y_1 + 2y_2$ and let $x_2 = z_1 + 2z_2$ (both have upper bounds of 3, so in the binary representation we only need two new variables for each to model the value possibilities of 0,1,2, and 3 for each variable.

Substituting this into the formulation given yields:

$$\begin{aligned} \texttt{Maximize:} \quad & -3y_1 - 6y_2 + 5z_1 + 10z_2 \\ \texttt{s.t.} \quad & 5y_1 + 10y_2 - 7z_1 - 14z_2 \geq 3 \\ & y_i \in \{0,1\} \quad \text{for } i = 1,2 \\ & z_i \in \{0,1\} \quad \text{for } i = 1,2 \end{aligned}$$

Solving the LP-relaxation of this yields the solution $(y_1, y_2, z_1, z_2) = (1, 1, 0, \frac{6}{7})$ with objective - $\frac{3}{7}$
Branch on z_2 to create two new subproblems $z_2 = 0$ yields the solution $(0, 1, 1, 0)$ with objective value -1. Adding $z_2 = 1$ creates infeasibility. We have therefore found the same solution as that found in parts a) and b). The solution $(0, 1, 1, 0)$ is equivalent to $x_1 = 2$ and $x_2 = 1$.

11.4 Exercise 4

River Power has four generators currently available for production and wishes to decide which to put on line to meet the expected 700-megawatt peak demand over the next several hours. The following table shows the cost to operate each generator (in thousands of dollars per hour) and their outputs in megawatts.

	Generator			
	1	2	3	4
Operating Cost ($000s)	7	12	5	14
Output Power (Megawatts)	300	600	500	1600

Table 7: Exercise 4 Data

(a) Formulate this problem as a BIP.

$$x_i = \begin{cases} 1 \text{ if generator } i \text{ is switched on, for } i = 1,2,3,4 \\ 0 \text{ otherwise} \end{cases}$$

$$\texttt{Minimize:} \quad 7x_1 + 12x_2 + 5x_3 + 14x_4 \tag{82}$$

$$\texttt{s.t.} \quad 300x_1 + 600x_2 + 500x_3 + 1600x_4 \geq 700 \tag{83}$$

$$x_i \in \{0,1\} \tag{84}$$

(b) Use the Branch-&-bound algorithm to find an optimal solution to this problem. **Hint:** Use an LP solver to solve all subproblems that arise.

Solve the LP relaxation: This yields the solution $(x_1, x_2, x_3, x_4) = (0, 0, 0, 0.44)$, with objective value 6.12. Set the upper bound to ∞

Branch on x_4.
Create two new subproblems, one with the new constraint $x_4 = 0$ (S1) and one with $x_4 = 1$ (S2)

Solve S1: This yields the solution $(x_1, x_2, x_3, x_4) = (0, 0.33, 1, 0)$, with objective value 9
Solve S2: This yields the solution $(x_1, x_2, x_3, x_4) = (0, 0, 0, 1)$, with objective value 14. This integer feasible, update upper bound to 14, and fathom

Look at S1 and branch on x_2. Create two new subproblems, one with the new constraint $x_2 = 0$ (S3) and one with $x_2 = 1$ (S4)

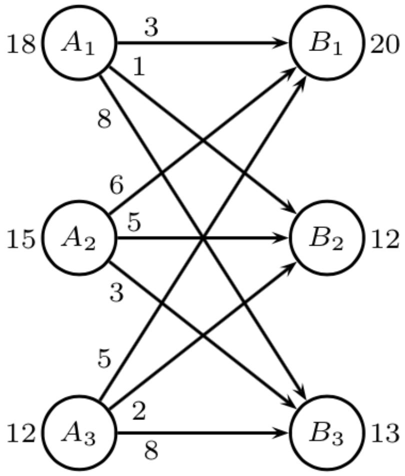
Solve S3: This yields the solution $(x_1, x_2, x_3, x_4) = (0.67, 0, 1, 0)$, with objective value 9.67
Solve S4: This yields the solution $(x_1, x_2, x_3, x_4) = (0, 1, 0.2, 0)$, with objective value 13

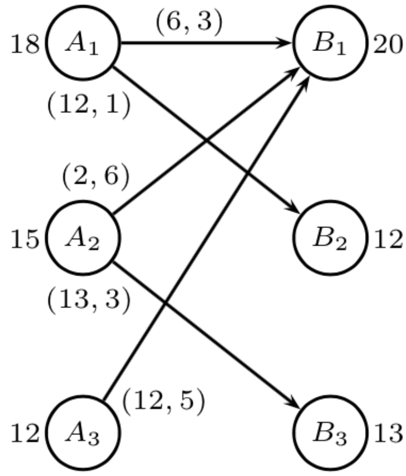
Continuing in this way will ultimately lead to the solution $(x_1^*, x_2^*, x_3^*, x_4^*) = (1, 0, 1, 0)$ with objective value 12.

12 Netwok simplex

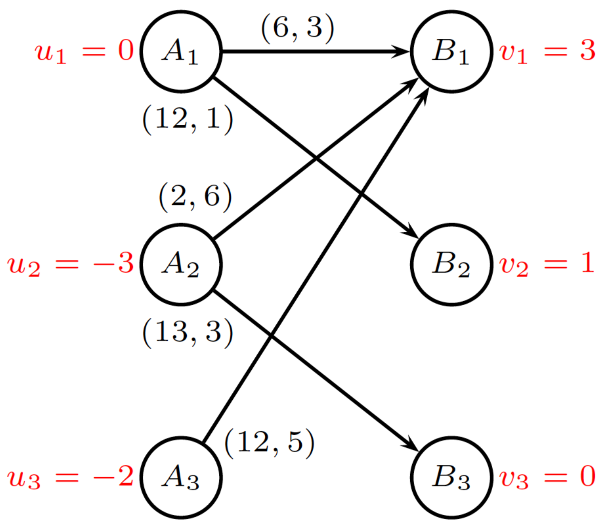
12.1 Exercise 1

When solving a relatively simple transportation problem it is always a good idea to visualize the problem. The right image shows the given but not optimal solution.





The dual variables are found by following the solutions edges around the problem. If you move forward through an edge you add the price of the edge to the dual variable and if you move backwards you subtract the cost of the edge. The dual variable of the starting point is always equal to zero, so we know that $u_1 = 0$. The cost to go from A_1 to B_1 is 3 so the dual variable v_1 becomes $v_1 = 0 + 3 = 3$. The price to go from B_1 to A_2 is 6 so the dual variable is $u_2 = 3 - 6 = -3$. Continuing in the same manner gives us the dual variables shown on the figure below which corresponds to answer 1.1B.



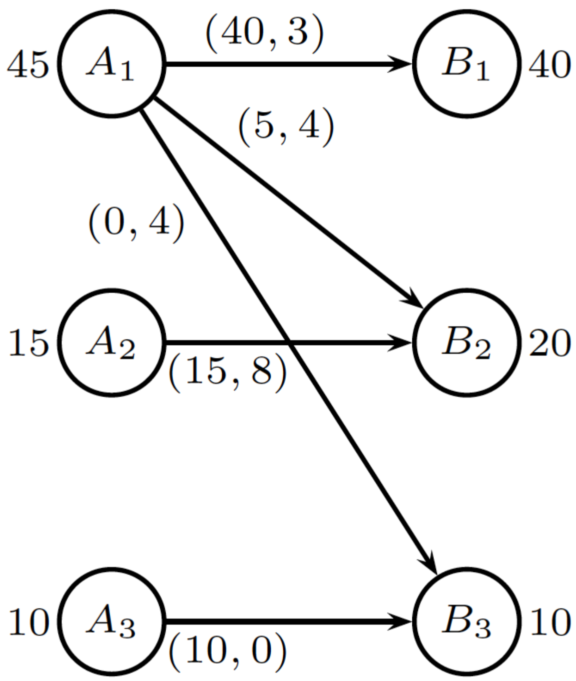
Moving on to the next question we have to find out which edge should be added in the next iteration of the network simplex. To do so the reduced cost for all of the unused edges are computed:

red cost (A1,B3): $c_{A1,B3} + u_1 - v_3 = 8 - 0 - 0 = 8$
red cost (A2,B2): $c_{A2,B2} + u_2 - v_2 = 5 - 3 - 1 = 1$
red cost (A3,B2): $c_{A3,B2} + u_3 - v_2 = 2 - 2 - 1 = -1$
red cost (A3,B3): $c_{A3,B3} + u_3 - v_3 = 8 - 2 - 0 = 6$

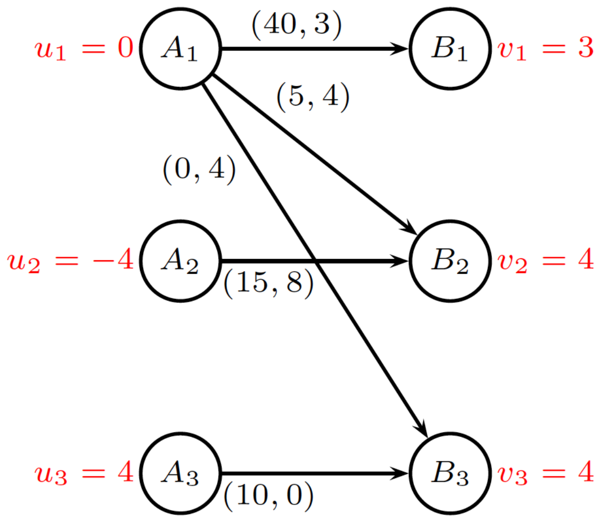
Only the edge (A3,B2) has a negative reduced cost so that is the edge we want to add to the solution. Now we have to determine the flow. Adding the edge we get the circle (A3,B2),(A1,B2),(A1,B1),(A3,B1). In order to determine the flow we have to look at the backwards flow in the circle i.e. the flow from (A1,B2) and (A3,B1). The flow in both of the edges is 12 so the flow in the new edge (A3,B2) should be 12 corresponding to answer 1.2A.

12.2 Exercise 2

The greedy algorithm gives the following solution. Notice that an edge with zero flow (A1,B3) is added in order to make the graph connected. A rule of thumb is that there always should be #nodes - 1 edges. The cost of the above initial solution is $40 \cdot 3 + 5 \cdot 4 + 0 \cdot 4 + 15 \cdot 8 + 10 \cdot 0 = 260$. In the graph below the flow and cost is written as (x,c), where x is the flow and c is the cost.

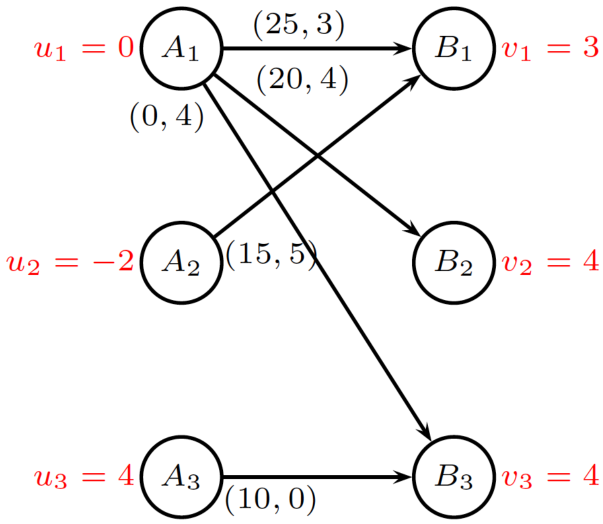


To check if the above graph is optimal we need to compute the reduced costs for all of the unused edges. First the dual variables are found by following the solutions edges around the problem. If you move forward through an edge you add the price of the edge to the dual variable and if you move backwards you subtract the cost of the edge. The dual variable of the starting point is always equal to zero, so we know that $u_1 = 0$. The cost to go from A_1 to B_2 is 4 so the dual variable v_2 becomes $v_2 = 0 + 4 = 4$. The price to go from B_2 to A_2 is 8 so the dual variable is $u_2 = 4 - 8 = -4$. Continuing in the same manner gives us the dual variables shown on the figure below.



The reduced costs are calculated
red cost (A2,B1): $c_{A2,B1} + u_2 - v_1 = 5 - 4 - 3 = -2$
red cost (A2,B3): $c_{A2,B3} + u_2 - v_3 = 8 - 4 - 4 = 0$
red cost (A3,B1): $c_{A3,B1} + u_3 - v_1 = 0 + 4 - 3 = 1$
red cost (A3,B2): $c_{A3,B2} + u_3 - v_2 = M + 4 - 4 = M$

It is seen that it will improve the solution to add the edge (A2,B1). Now we have to determine the flow of the new edge. When adding the edge we get the circle (A1,B2), (A2,B2), (A2,B1), (A1,B1). The backward flow in this circle is (A2,B2) and (A1,B1) with a backwards flow at 15 and 40. 15 is the smallest flow so we'll remove the edge (A2,B2) and the new edge (A2,B1) will get a flow at 15. We can now compute the dual variables in the same manner as before for our new graph.



Again to check if the solution is optimal we compute the reduced costs for the unused edges.

red cost (A2,B2): $c_{A2,B2} + u_2 - v_2 = 8 - 2 - 4 = 2$

red cost (A2,B3): $c_{A2,B3} + u_2 - v_3 = 8 - 2 - 4 = 2$
red cost (A3,B1): $c_{A3,B1} + u_3 - v_1 = 0 + 4 - 3 = 1$
red cost (A3,B2): $c_{A3,B2} + u_3 - v_2 = M + 4 - 4 = M$

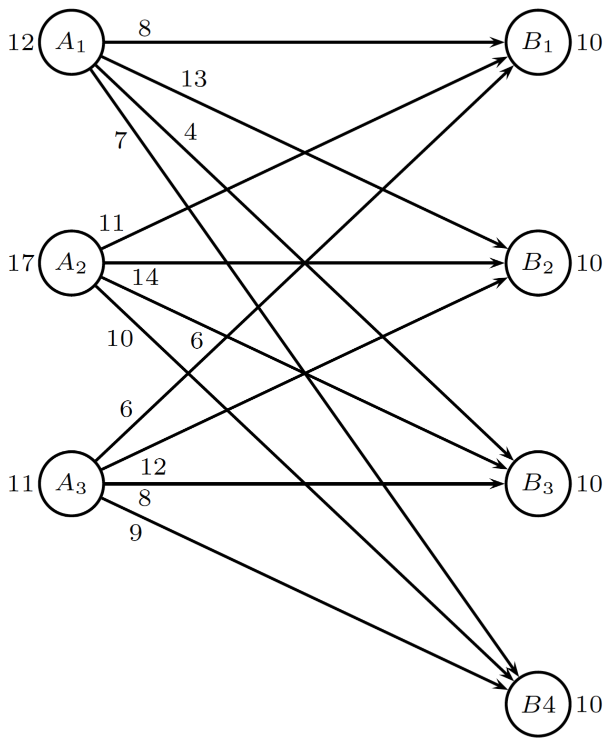
Since all of the reduced costs are greater than zero the found solution is optimal. The cost of the optimal solution is $25 \cdot 3 + 20 \cdot 4 + 0 \cdot 4 + 15 \cdot 5 + 10 \cdot 0 = 230$.

12.3 Exercise 3

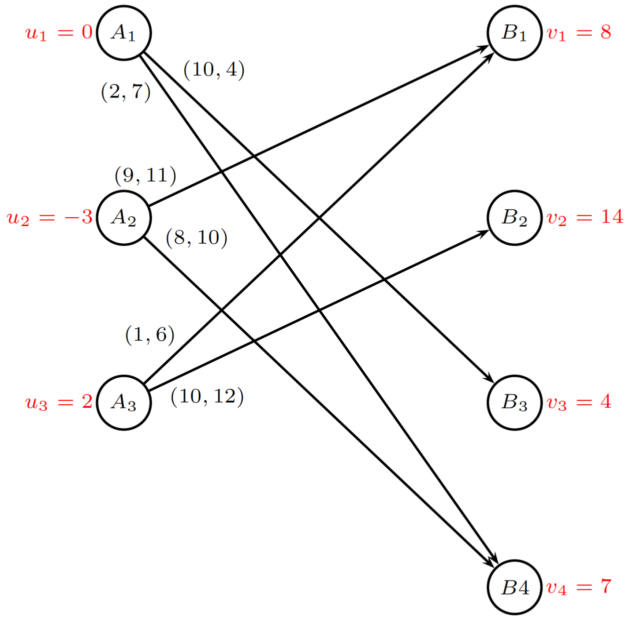
To minimize the total shipping cost it can be shown that we only have to minimize the miles. By changing the unit from miles to hecto-miles we get the following parameter table:

		Distribution center				
		1	2	3	4	**Supply**
	1	8	13	4	7	12
Plant	2	11	14	6	10	17
	3	6	12	8	9	11
Demand		10	10	10	10	

This can be drawn as the following network representation:



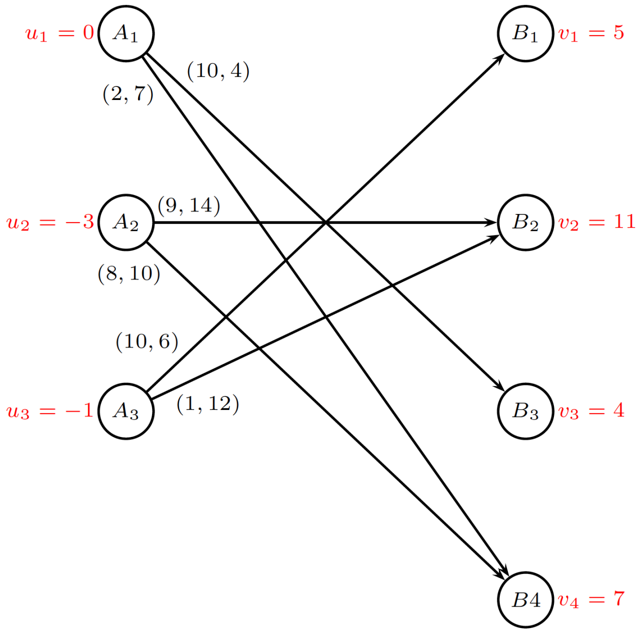
In order to obtain the optimal solution we start out with the greedy solution shown below and compute all of the dual variables as shown in earlier exercises.



To check if the solution is optimal we compute the reduced costs for the unused edges.

red cost (A1,B1): $c_{A1,B1} + u_1 - v_1 = 8 + 0 - 8 = 0$
red cost (A1,B2): $c_{A1,B2} + u_1 - v_2 = 13 + 0 - 14 = -1$
red cost (A2,B2): $c_{A2,B2} + u_2 - v_2 = 14 - 3 - 14 = -3$
red cost (A2,B3): $c_{A2,B3} + u_2 - v_3 = 6 - 3 - 4 = -1$
red cost (A3,B3): $c_{A3,B3} + u_3 - v_3 = 8 + 2 - 4 = 6$
red cost (A3,B4): $c_{A3,B4} + u_3 - v_4 = 9 + 2 - 7 = 4$

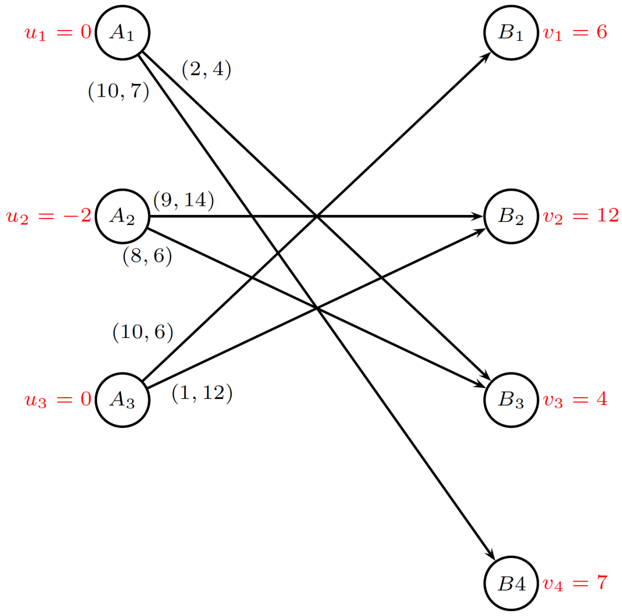
The most negative reduced cost is from (A2,B2). When adding the edge (A2,B2) we get the circle (A2,B2), (A3,B2), (A3,B1), (A2,B1). The backward flow in this circle is (A3,B2) and (A2,B1) with a backwards flow at 10 and 9. 9 is the smallest flow so we'll remove the edge (A2,B1) and the new edge (A2,B2) will get a flow at 9. We can now compute the dual variables in the same manner as before for our new graph.



Again the reduced costs are computed:

red cost (A1,B1): $c_{A1,B1} + u_1 - v_1 = 8 + 0 - 5 = 3$
red cost (A1,B2): $c_{A1,B2} + u_1 - v_2 = 13 + 0 - 11 = 2$
red cost (A2,B1): $c_{A2,B1} + u_2 - v_1 = 11 - 3 - 5 = 3$
red cost (A2,B3): $c_{A2,B3} + u_2 - v_3 = 6 - 3 - 4 = -1$
red cost (A3,B3): $c_{A3,B3} + u_3 - v_3 = 8 - 1 - 4 = 3$
red cost (A3,B4): $c_{A3,B4} + u_3 - v_4 = 9 - 1 - 7 = 1$

Since the reduced cost from A2 to B3 is negative the solution is still not optimal. By adding the edge (A2,B3) we get the circle (A2,B3), (A1,B3), (A1,B4), (A2, B4). The backward flow in this circle is (A1,B3) and (A2,B4) with a backwards flow at 10 and 8. 8 is the smallest flow so we'll remove the edge (A2,B4) and the new edge (A2,B3) will get a flow at 8. We can now compute the dual variables in the same manner as before for our new graph.



To see if the found solution is optimal we again compute the reduced costs for the unused edges.

red cost (A1,B1): $c_{A1,B1} + u_1 - v_1 = 8 + 0 - 6 = 2$
red cost (A1,B2): $c_{A1,B2} + u_1 - v_2 = 13 + 0 - 12 = 1$
red cost (A2,B1): $c_{A2,B1} + u_2 - v_1 = 11 - 2 - 6 = 3$
red cost (A2,B4): $c_{A2,B4} + u_2 - v_4 = 10 - 2 - 7 = 1$
red cost (A3,B3): $c_{A3,B3} + u_3 - v_3 = 8 + 0 - 4 = 4$
red cost (A3,B4): $c_{A3,B4} + u_3 - v_4 = 9 + 0 - 7 = 2$

Since all of the reduced costs are greater than zero the found solution is optimal. The cost of the optimal solution is $2 \cdot 4 + 10 \cdot 7 + 9 \cdot 14 + 8 \cdot 6 + 10 \cdot 6 + 1 \cdot 12 = 324$.

12.4 Exercise 4

From the information given in the text we know that there is three products and five plants. Plant one, two and three can produce all of the three products but plant four and five can only produce product one and two. In order to formulate this as a transportation problem we say assume that plant four and five can produce all of the three products but the cost of producing product three at these plants are M i.e. a really big number. With the information given we can construct the following parameter table:

		Product			
		1	2	3	**Supply**
	1	41	55	48	40
	2	39	51	45	60
Plant	3	42	56	50	40
	4	38	52	M	60
	5	39	53	M	100
Demand		70	100	90	260/300

It is seen that the supply and demand is not equal so in order to use network simplex to solve the problem we need to introduce a dummy product with demand 40 to make the problem equal in demand and supply. Be doing this we get the following parameter table.

		Product				
		1	2	3	Dummy	**Supply**
	1	41	55	48	0	40
	2	39	51	45	0	60
Plant	3	42	56	50	0	40
	4	38	52	M	0	60
	5	39	53	M	0	100
Demand		70	100	90	40	300/300

Note: You are NOT asked to solve the problem, only to formulate the above parameter table.

13 Linear Programming: Dual Simplex

13.1 Exercise 5

Normally to solve a problem like this we would need to use the 2-phase method, however dual-simplex allows us to solve the problem directly. In dual-simplex we still need all the constraints to be on the $\leq$-form but we also need a negative right hand side, so the first step is to multiply the two constraints with -1 resulting in the new problem.

$$
\begin{aligned}
\text{min} \qquad & Z = 5x_1 + 2x_2 + 4x_3 \\
\text{s.t.} \qquad & -3x_1 - x_2 - 2x_3 \leq -4 \\
& -6x_1 - 3x_2 - 5x_3 \leq -10 \\
& x_1 \geq 0, x_2 \geq 0, x_3 \geq 0
\end{aligned}
$$

In tabular form this is

Bv	Z	x_1	x_2	x_3	x_4	x_5	RHS
Z	-1	5	2	4	0	0	0
x_4	0	-3	-1	-2	1	0	-4
x_5	0	-6	-3	-5	0	1	-10

In dual simplex the outgoing variable is first found as the variable with the most negative right hand side. So in the above case x_5 is the leaving variable. The in going variable is like in normal simplex found by a minimum ratio test, but only the negative variables in the pivot row are considered. Here we have $\frac{5}{6}$, $\frac{2}{3}$ and $\frac{4}{5}$. $\frac{2}{3}$ is smallest so x_2 is the in going variable.

Next step is like in normal simplex to get a 1 in the pivot element and the rest of the pivot column should be zero. This is done by the following operations:
$R2 = R2/(-3) \rightarrow R0 = R0 - 2R2 \rightarrow R1 = R1 + R2$, resulting in the next table

Bv	Z	x_1	x_2	x_3	x_4	x_5	RHS
Z	-1	1	0	2/3	0	2/3	-20/3
x_4	0	-1	0	-1/3	1	-1/3	-2/3
x_2	0	2	1	5/3	0	-1/3	10/3

Here x_4 is the leaving variable and the in going is again found with at min ratio test. The ratios are: $\frac{1}{1}$, $\frac{2/3}{1/3}$ and $\frac{2/3}{1/3}$. The minimum ratio is $\frac{1}{1}$ so x_1 is the in going variable.

To get to the next iteration the following row operations have been done
$R1 = R1 \cdot (-1) \rightarrow R0 = R0 - R1 \rightarrow R2 = R2 - 2R1$, resulting in the final table

Bv	Z	x_1	x_2	x_3	x_4	x_5	RHS
Z	-1	0	0	1/3	1	1/3	-22/3
x_1	0	1	0	1/3	-1	1/3	2/3
x_2	0	0	1	1	2	-1	2

The optimal solution to the problem is found to $Z^* = \frac{22}{3}$, when $(x_1, x_2) = (\frac{2}{3}, 2)$.

13.2 Exercise 6

First of all the problem is written in standard form by multiplying the second constraint with -1

$$\max \quad Z + x_1 + x_2 = 0 \tag{85}$$

$$\text{s.t.} \quad x_1 + x_2 \leq 8 \tag{86}$$

$$-x_2 \leq -3 \tag{87}$$

$$-x_1 + x_2 \leq 2 \tag{88}$$

The dual simplex method is applied to solve the problem sine the right hand side is negative.

	Z	x_1	x_2	x_3	x_4	x_5	RHS
Z	1	1	1	0	0	0	0
x_3	0	1	1	1	0	0	8
x_4	0	0	-1	0	1	0	-3
x_5	0	-1	1	0	0	1	2

x_4 is the leaving basis variable and x_2 is the entering basis variable since it is the only negative coefficient in the row. The tabular is then made legal with x_2 as a basis variable.

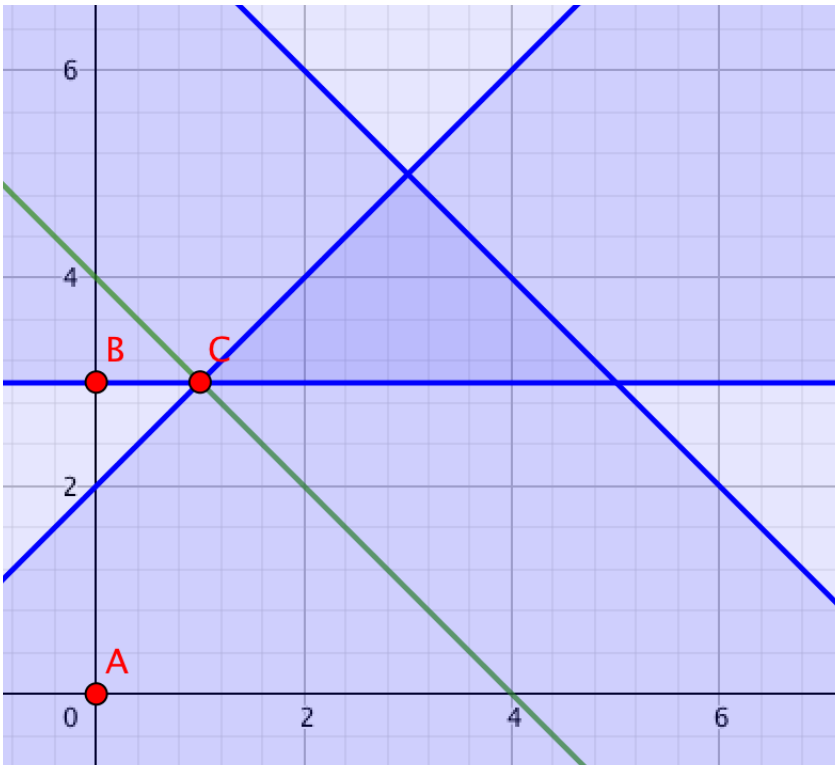
	Z	x_1	x_2	x_3	x_4	x_5	RHS
Z	1	1	0	0	1	0	-3
x_3	0	1	0	1	1	0	5
x_2	0	0	1	0	-1	0	3
x_5	0	-1	0	0	1	1	-1

The right hand side is still negative, so another iteration of the dual simplex method is made. x_5 is the leaving basis variable while x_1 is the entering basis variable.

	Z	x_1	x_2	x_3	x_4	x_5	RHS
Z	1	0	0	0	2	1	-4
x_3	0	0	0	1	2	1	4
x_2	0	0	1	0	-1	0	3
x_1	0	1	0	0	-1	-1	1

The optimal solution is $x_1 = 1$, $x_2 = 3$ and $Z = -4$

The graphical solution can be seen below. The darkest blue triangle is the feasible solution space and the green line is the objective function. The dual simplex method start in point A(0;0). After the first iteration it reaches point B(0;3) and in the final iteration it reaches the optimal point C(1;3). The difference between the dual simplex and the standard simplex is that the dual simplex visit illegal solutions before it reaches the optimal legal CP-solution. The standard simplex method visit only legal CP-solutions before it reaches the optimal legal CP-solution.


6
4
2
0
2
4
6
A
B
C

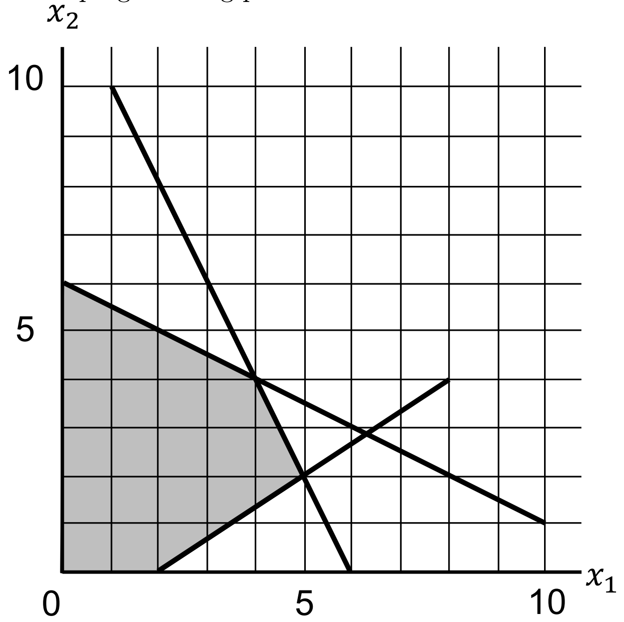
14 Exam F18

Written exam F18
Course name: Introduction to Operations Research
Course number: 42101
Allowed aids: All aids and materials permitted. Use of internet is not allowed. Answers to multiple choice questions are indicated on the form below. Answers to multiple choice tasks can also be handed in on exam papers if you make a mistake on the form. Remember to write study number of all pages you hand in.
Duration: 4 hours
Weights of questions: The exam consists of 17 questions. Question 10 and 16 are text-based questions and a full answer including explanations should be handed in. All other questions are multiple-choice questions. The text Question 10 count for 15 points and question 16 count for 10 points. Each correct answer to a multiple choice questions gives 5 point. A wrong answers give 0 points. Only one choice is allowed per multiple choice question. If more than one choice is selected for a multiple choice questions then the answer is counted as wrong. The maximum number of points are 100.

Exercise	Answer					
1	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
2	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
3	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
4	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
5	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
6	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
7	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
8	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
9	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
10	Text question					
11	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
12	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
13	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
14	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
15	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐
16	Text question					
17	A: ☐	B: ☐	C: ☐	D: ☐	E: ☐	F: ☐

Name:
Student number:

Exercise 1: Linear programming The below figure shows the space of legal solutions (gray area) to a linear programming problem.



Which combination of constraints describes the problem best.

1A)

$$\begin{aligned} -\frac{2}{3}x_1 - x_2 &\leq \frac{4}{3} \\ \frac{1}{2}x_1 - x_2 &\leq 6 \\ 2x_1 + x_2 &\leq 8 \\ x_1, x_2 &\geq 0 \end{aligned}$$

1B)

$$\begin{aligned} \frac{2}{3}x_1 - x_2 &\geq \frac{4}{3} \\ \frac{1}{2}x_1 + x_2 &\leq 6 \\ 2x_1 - x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

1C)

$$\begin{aligned} \frac{2}{3}x_1 - x_2 &\leq \frac{4}{3} \\ \frac{1}{2}x_1 + x_2 &\leq 6 \\ 2x_1 + x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

1D)

$$\begin{aligned} -\frac{2}{3}x_1 - x_2 &\leq \frac{4}{3} \\ \frac{1}{2}x_1 + x_2 &\leq 6 \\ 2x_1 - x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

1E)

$$\begin{aligned} -\frac{2}{3}x_1 - x_2 &\geq \frac{4}{3} \\ \frac{1}{2}x_1 + x_2 &\leq 6 \\ 2x_1 - x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

1F)

$$\begin{aligned} -\frac{2}{3}x_1 - x_2 &\leq \frac{4}{3} \\ \frac{1}{2}x_1 + x_2 &\leq 4 \\ 2x_1 + x_2 &\leq 4 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Exercise 2: Simplex Still considering the problem from Exercise 1. Assume the objective function for the problem is

$$\max 4x_1 + 5x_2$$

Which sequence of solutions (x_1, x_2) will the simplex algorithm then visit?

2A)
$$(0,0) \to (0,6)$$

2B)
$$(0,0) \to (2,0) \to (5,2) \to (4,4)$$

2C)
$$(0,0) \to (0,6) \to (4,4)$$

2D)
$$(0,0) \to (2,0)$$

2E)
$$(0,0) \to (2,0) \to (5,2)$$

2F)
$$(0,0) \to (0,6) \to (4,4) \to (5,2)$$

Exercise 3: Simplex Consider the following table:

b.v.	eq.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	0	-9	-7	0	0	3	36
x4	1	0	0	2	1	1	0	-1	8
x5	2	0	0	0	3	0	1	$-\frac{1}{3}$	10
x1	3	0	1	0	0	0	0	$\frac{1}{3}$	4

Complete one iteration of the simplex algortihm on the above table. How will the table look after the iteration?

3A)

b.v.	e.q.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	-9	-9	-7	0	0	0	0
x4	1	0	3	2	1	1	0	0	20
x5	2	0	1	0	3	0	1	0	14
x6	3	0	3	0	0	0	0	1	12

3B)

b.v.	e.q.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	$\frac{9}{2}$	0	$-\frac{5}{2}$	$\frac{9}{2}$	0	0	90
x2	1	0	$\frac{3}{2}$	1	$\frac{1}{2}$	$\frac{1}{2}$	0	0	10
x5	2	0	1	0	3	0	1	0	14
x6	3	0	3	0	0	0	0	1	12

3C)

b.v.	e.q.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	$-\frac{20}{3}$	-9	0	0	$\frac{8}{3}$	0	$\frac{98}{3}$
x4	1	0	$\frac{8}{3}$	2	0	1	$-\frac{1}{3}$	0	$\frac{46}{3}$
x3	2	0	$\frac{1}{3}$	0	1	0	$\frac{1}{3}$	0	$\frac{14}{3}$
x6	3	0	3	0	0	0	0	1	12

3D)

b.v.	e.q.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	0	-9	0	0	$\frac{7}{3}$	$\frac{20}{9}$	$\frac{178}{3}$
x4	1	0	0	2	0	1	$-\frac{1}{3}$	$-\frac{8}{9}$	$\frac{14}{3}$
x3	2	0	0	0	1	0	$\frac{1}{3}$	$-\frac{1}{9}$	$\frac{10}{3}$
x1	3	0	1	0	0	0	0	$\frac{1}{3}$	4

3E)

b.v.	e.q.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	0	0	0	$\frac{9}{2}$	$\frac{5}{6}$	$-\frac{16}{9}$	$\frac{241}{3}$
x2	1	0	0	1	0	$\frac{1}{2}$	$-\frac{1}{6}$	$-\frac{4}{9}$	$\frac{7}{3}$
x3	2	0	0	0	1	0	$\frac{1}{3}$	$-\frac{1}{9}$	$\frac{10}{3}$
x1	3	0	1	0	0	0	0	$\frac{1}{3}$	4

3F)

b.v.	e.q.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	0	0	$-\frac{5}{2}$	$\frac{9}{2}$	0	$-\frac{3}{2}$	72
x2	1	0	0	1	$\frac{1}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$	4
x5	2	0	0	0	3	0	1	$-\frac{1}{3}$	10
x1	3	0	1	0	0	0	0	$\frac{1}{3}$	4

Exercise 4: Fundamental insight Consider the following LP

$$\max 9x_1 + 8x_2 + 2x_3$$

$$1x_1 + 2x_2 + 1x_3 \leq 12 \tag{89}$$
$$2x_1 + 2x_2 + 1x_3 \leq 15 \tag{90}$$
$$1x_1 + 1x_2 + 1x_3 \leq 7 \tag{91}$$
$$x_1, x_2, x_3 \geq 0$$

When the problem is written in augmented form the slack variables x_4, x_5 and x_6 are added corresponding to the three constraints (89), (90) and (91).
Let x_1, x_2 and x_5 be in the basis. How will the table corresponding to this choice of basis variables look?

4A)

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	0	1	7	0	0	9	63
x_4	1	0	0	1	0	1	0	-1	5
x_5	2	0	0	0	-1	0	1	-2	1
x_1	3	0	1	1	1	0	0	1	7

4B)

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	0	1	7	0	0	9	63
x_1	1	0	0	1	0	1	0	-1	5
x_2	2	0	0	0	-1	0	1	-2	1
x_5	3	0	1	1	1	0	0	1	7

4C)

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	0	0	0	-1	-2	9	63
x_1	1	0	1	0	0	1	0	-1	5
x_2	2	0	0	1	0	0	1	-2	1
x_5	3	0	0	0	1	3	4	1	7

4D)

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	0	0	7	-1	0	10	58
x_1	1	0	1	0	1	-1	0	2	2
x_2	2	0	0	1	0	1	0	-1	5
x_5	3	0	0	0	-1	0	1	-2	1

4E)

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	0	0	0	-1	7	-4	65
x_1	1	0	1	0	0	-1	1	0	3
x_2	2	0	0	1	0	1	0	-1	5
x_5	3	0	0	0	1	0	-1	2	-1

4F)

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	1	2	7	0	0	0	59
x_1	1	0	2	2	1	1	0	0	3
x_2	2	0	0	3	0	0	1	0	5
x_5	3	0	3	0	-1	0	0	1	2

Exercise 5: Duality Now consider the following LP

$$\min 2x_1 + 6x_2 + 3x_3$$

$$x_1 + 4x_2 + 2x_3 = 6 \tag{92}$$
$$2x_1 + 2x_2 + 3x_3 \leq 7 \tag{93}$$
$$1x_1 + 2x_2 + 1x_3 \geq 1 \tag{94}$$
$$x_1 \geq 0,\ x_2 \in \mathbb{R},\ x_3 \leq 0 \tag{95}$$

In the dual problem we use the variables y_1, y_2 and y_3 (corresponding to the constraints (92), (93) and (94)) What will the dual problem be?

5A)

$$\max 6y_1 + 7y_2 + y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\geq 2 \\ 4y_1 + 2y_2 + 2y_3 &= 6 \\ 2y_1 + 3y_2 + y_3 &\leq 3 \\ y_1 \in \mathbb{R}, y_2 \geq 0, y_3 &\leq 0 \end{aligned}$$

5B)

$$\max 6y_1 + 7y_2 + y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\leq 2 \\ 4y_1 + 2y_2 + 2y_3 &\leq 6 \\ 2y_1 + 3y_2 + y_3 &\leq 3 \\ y_1 \geq 0, y_2 \geq 0, y_3 &\geq 0 \end{aligned}$$

5C)

$$\min 6y_1 + 7y_2 + y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\geq 2 \\ 4y_1 + 2y_2 + 2y_3 &= 6 \\ 2y_1 + 3y_2 + y_3 &\leq 3 \\ y_1 \in \mathbb{R}, y_2 \geq 0, y_3 &\leq 0 \end{aligned}$$

5D)

$$\max 6y_1 + 7y_2 + y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\leq 2 \\ 4y_1 + 2y_2 + 2y_3 &= 6 \\ 2y_1 + 3y_2 + y_3 &\geq 3 \\ y_1 \in \mathbb{R}, y_2 \leq 0, y_3 &\geq 0 \end{aligned}$$

5E)

$$\max 2y_1 + 6y_2 + 3y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\leq 6 \\ 4y_1 + 2y_2 + 2y_3 &\leq 7 \\ 2y_1 + 3y_2 + y_3 &\leq 1 \\ y_1 \geq 0, y_2 \geq 0, y_3 &\geq 0 \end{aligned}$$

5F)

$$\max 6y_1 + 7y_2 + y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\leq 2 \\ 4y_1 + 2y_2 + 2y_3 &= 6 \\ 2y_1 + 3y_2 + y_3 &\leq 3 \\ y_1 \geq 0,\ y_2 \in \mathbb{R},\ y_3 &\leq 0 \end{aligned}$$

Exercise 6: Duality Consider two LP problems in matrix form (x is a column vector with decision variables)

LP1:

$$\max Z_1 = cx$$

subject to

$$\begin{aligned} A_1 x &\leq b_1 \\ x &\geq 0 \end{aligned}$$

LP2:

$$\max Z_2 = cx$$

subject to

$$\begin{aligned} A_2 x &\leq b_2 \\ x &\geq 0 \end{aligned}$$

The two problems have an equal amount of variables and the objective functions are identical. The constraints to the two problems are not the same (meaning $A_1 \neq A_2$ and $b_1 \neq b_2$). It is know that the optimal solution to LP1 is $Z_1^* = 42$ and that every solution x fulfilling the constraints in LP1 also fulfills the constraints in LP2. What is most correct to say about the solution to LP2's dual problem?

6A) The dual problem to LP2 might not have a feasible solution. If there is a feasible solution the objective value will at max be 42.

6B) The dual problem to LP2 will have the objective value 42.

6C) The dual problem to LP2 is unconstrained.

6D) The dual problem to LP2 has no feasible solution.

6E) The dual problem to LP2 might not have a feasible solution. If there is a feasible solution the objective value is at least 42.

6F) None of the answers 6A-6E are correct.

Exercise 7: Sensitivity analysis Consider the following LP

$$\max Z = 7x_1 + 8x_2 + 10x_3$$

$$2x_1 + 3x_2 + 2x_3 \leq 1 \tag{96}$$
$$2x_1 + 1x_2 \leq 18 \tag{97}$$
$$3x_1 + 4x_2 + 2x_3 \leq 2 \tag{98}$$
$$x_1, x_2, x_3 \geq 0$$

When rewritten to augmented form we add three slack variables x_4, x_5 and x_6 corresponding to the three constraints(96)–(98).

In the optimal solution x_3, x_5 and x_6 are basis variables.
We want to determine how the optimal solution change when the right hand side in constraint (98) change. The constraint is rewritten to

$$3x_1 + 4x_2 + 2x_3 \leq 2 + \Delta.$$

In which interval does the basis remain the same?
How does the objective value change in this interval?

7A) The interval where the basis will remain the same is $\Delta \in [-\infty; 3]$ and $Z^* = 5 + \Delta$ in this interval

7B) The interval where the basis will remain the same is $\Delta \in [-1; \frac{5}{3}]$ and $Z^* = 5 + \Delta$ in this interval

7C) The interval where the basis will remain the same is $\Delta \in [-\frac{5}{3}; \infty]$ and $Z^* = 5 + 5\Delta$ in this interval

7D) The interval where the basis will remain the same is $\Delta \in [-\frac{5}{3}; \infty]$ and $Z^* = 5$ in this interval

7E) The interval where the basis will remain the same is $\Delta \in [-1; \infty]$ and $Z^* = 5 + 5\Delta$ in this interval

7F) The interval where the basis will remain the same is $\Delta \in [-1; \infty]$ and $Z^* = 5$ in this interval

Exercise 8 - Sensitivity analysis - the objective function Consider the LP from Exercise 7:

$$\max Z = 7x_1 + 8x_2 + 10x_3$$

$$2x_1 + 3x_2 + 2x_3 \leq 1 \quad (99)$$
$$2x_1 + 1x_2 \leq 18 \quad (100)$$
$$3x_1 + 4x_2 + 2x_3 \leq 2 \quad (101)$$
$$x_1, x_2, x_3 \geq 0$$

When the problem is written at augmented form we add the slackvariables x_4, x_5 and x_6 corresponding to the three constraints (99)–(101). In the optimal solution x_3, x_5 and x_6 are basis variables and the optimal solution is $Z^* = 5$. We wish to investigate how the objective value changes if we instead consider the objective function

$$\max Z = 7x_1 + (8 + \Delta)x_2 + (10 - \Delta)x_3$$

here Δ is a variable. In which interval will $[x_3, x_5, x_6]$ remain in basis and how does the objective value change as function of Δ in this interval?

8A) The interval where the basis will remain the same is $\Delta \in [-\infty; 3]$ and $Z^* = 5 - 0,5\Delta$ in this interval

8B) The interval where the basis will remain the same is $\Delta \in [-\infty; \frac{14}{5}]$ and $Z^* = 5 - 0,5\Delta$ in this interval

8C) The interval where the basis will remain the same is $\Delta \in [-\infty; \frac{14}{3}]$ and $Z^* = 5 - 0,5\Delta$ in this interval

8D) The interval where the basis will remain the same is $\Delta \in [-\infty; 3]$ and $Z^* = 5 - \Delta$ in this interval

8E) The interval where the basis will remain the same is $\Delta \in [-\infty; \frac{14}{5}]$ and $Z^* = 5 - \Delta$ in this interval

8F) The interval where the basis will remain the same is $\Delta \in [-\infty; \frac{14}{3}]$ and $Z^* = 5 - \Delta$ in this interval

Exercise 9: Simplex, 2-phase method (text question, 15 points) Consider the following LP:

$$\max 7x_1 + 3x_2 + 1x_3$$

subject to

$$3x_1 + 2x_2 \geq 2$$
$$4x_1 + 2x_2 + 5x_3 \leq 7$$
$$1x_1 + 4x_2 + 2x_3 \leq 14$$
$$x_1, x_2, x_3 \geq 0$$

Solve the LP using the 2-phase method. Show all steps and explain your computations.

Exercise 10: Transportation problem In this exercise we will consider a transportation problem where goods need to be shipped from 4 factories to 2 stores. We will denote the factories A1, A2, A3, A4. And the stores B1, and B2. In the table below the shipping costs per unit between each pair of factory/store can be seen. The supply from each factory and demand from each store can also be seen.

	B1	B2	Supply
A1	1	2	20
A2	9	8	18
A3	5	7	15
A4	2	6	12
Demand	30	35	

A legal but not optimal solution can be seen below. The table shows how many units that are shipped between factories and stores to fulfill the demand.

From	A1	A2	A3	A3	A4
To	B1	B2	B1	B2	B2
Amount	20	18	10	5	12

In this exercise you need to complete one iteration of network simplex as described in the note. Start with the above solution and see if it can be improved. Compute the reduced cost for the three edges which are not part of the above solution. Below $\bar{c}_{A1,B2}$ denotes the reduced costs for the edge $(A1, B2)$ and so on. Which of the following options are correct?

10A)

$$\bar{c}_{A1,B2} = 1, \bar{c}_{A2,B1} = -3, \bar{c}_{A4,B1} = 2$$

10B)

$$\bar{c}_{A1,B2} = -1, \bar{c}_{A2,B1} = 3, \bar{c}_{A4,B1} = 2$$

10C)

$$\bar{c}_{A1,B2} = -1, \bar{c}_{A2,B1} = 3, \bar{c}_{A4,B1} = -2$$

10D)

$$\bar{c}_{A1,B2} = -1, \bar{c}_{A2,B1} = -3, \bar{c}_{A4,B1} = 2$$

10E)

$$\bar{c}_{A1,B2} = 0, \bar{c}_{A2,B1} = -3, \bar{c}_{A4,B1} = 2$$

10F)

$$\bar{c}_{A1,B2} = 0, \bar{c}_{A2,B1} = -1, \bar{c}_{A4,B1} = -2$$

Exercise 11: Transportation problem Since at least one of the edges has a negative reduced cost we need to determine which of the edges enters the basis and which of the edges leaves the basis. How will the solution look **after one iteration of network simplex**?

10A)

From	A1	A2	A2	A3	A4
To	B1	B1	B2	B2	B2
Amount	20	10	8	15	12

10B)

From	A1	A2	A2	A3	A4
To	B1	B1	B2	B2	B1
Amount	20	-2	20	15	12

10C)

From	A1	A2	A3	A3	A4
To	B2	B1	B1	B2	B2
Amount	20	18	12	3	12

10D)

From	A1	A2	A2	A3	A4
To	B2	B1	B2	B1	B1
Amount	20	3	15	15	3

10E)

From	A1	A2	A3	A4	A4
To	B1	B2	B2	B1	B2
Amount	20	18	15	10	2

10F)

From	A1	A2	A3	A4
To	B2	B1	B2	B1
Amount	20	18	15	12

Exercise 12: Assignment problem A company has four tasks (O1,O2,O3,O4) they need so solve. There is 5 employees in the company (P1, P2,. . .,P5). The employees are not equally good at each of the tasks. In the table below the cost of assigning each of the four tasks to each of the employees are shown. Each employee can at max solve one task and all of the tasks needs to be solved.

	O1	O2	O3	O4
P1	4	9	9	9
P2	7	5	5	7
P3	6	4	9	8
P4	1	9	7	5
P5	7	5	3	5

What is the objective value of the solution that is obtained by a greedy algorithm which, at any iteration, makes the cheapest person to task assignment available? If necessary, break ties by choosing the person with the lowest index.

12A) 11

12B) 12

12C) 13

12D) 14

12E) 15

12F) 16

Exercise 13: Integer programming A bakery bakes two types of bread. Bagels and rolls. The two essential ingredients in the breads are milk and flour. To bake one bread the following amount is needed:

	Bagels	Rolls
Flour	700 g	500 g
Milk	4 dl	2,5 dl

The bakery have 50 kg of flour and 20 liters of milk. The first 15 rolls sold can be sold for 20 DKK a piece. If the bakery produce more than 15 rolls the rest can be sold for 17 DKK a piece. Bagels can be sold to a price of 30 DKK no matter how many they produce. The bakery wants to build a model to maximize their income. The variable x_1 denotes the amount of bagels and x_2 the amount of rolls produced. y is a support variable $\mathbb{Z}_+$ is the non negative integers. Which of the following models is correct?

13A)

$$\max 30x_1 + 18.5x_2$$

subject to

$$\begin{aligned} 0{,}7x_1 + 0{,}5x_2 &\leq 50 \\ 0{,}4x_1 + 0{,}25x_2 &\leq 20 \\ x_2 + y &\geq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

13B)

$$\max 30x_1 + 20x_2 + 15y$$

subject to

$$\begin{aligned} 0{,}7x_1 + 0{,}5x_2 &\leq 50 \\ 0{,}4x_1 + 0{,}25x_2 &\leq 20 \\ x_2 &\leq 15 \\ y &\geq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

13C)

$$\max 30x_1 + 20x_2 - 17y$$

subject to

$$\begin{aligned} 0{,}7x_1 + 0{,}5x_2 &\leq 50 \\ 0{,}4x_1 + 0{,}25x_2 &\leq 20 \\ x_2 - y &\leq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

13D)

$$\max 30x_1 + 20x_2 + 17y$$

subject to

$$\begin{aligned} 0{,}7x_1 + 0{,}5x_2 &\leq 50 \\ 0{,}4x_1 + 0{,}25x_2 &\leq 20 \\ x_2 + y &\leq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

13E)

$$\max 30x_1 + 20x_2 + 17y$$

subject to

$$\begin{aligned} 0{,}7x_1 + 0{,}5x_2 &\leq 50 \\ 0{,}4x_1 + 0{,}25x_2 &\leq 20 \\ x_2 - y &\leq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

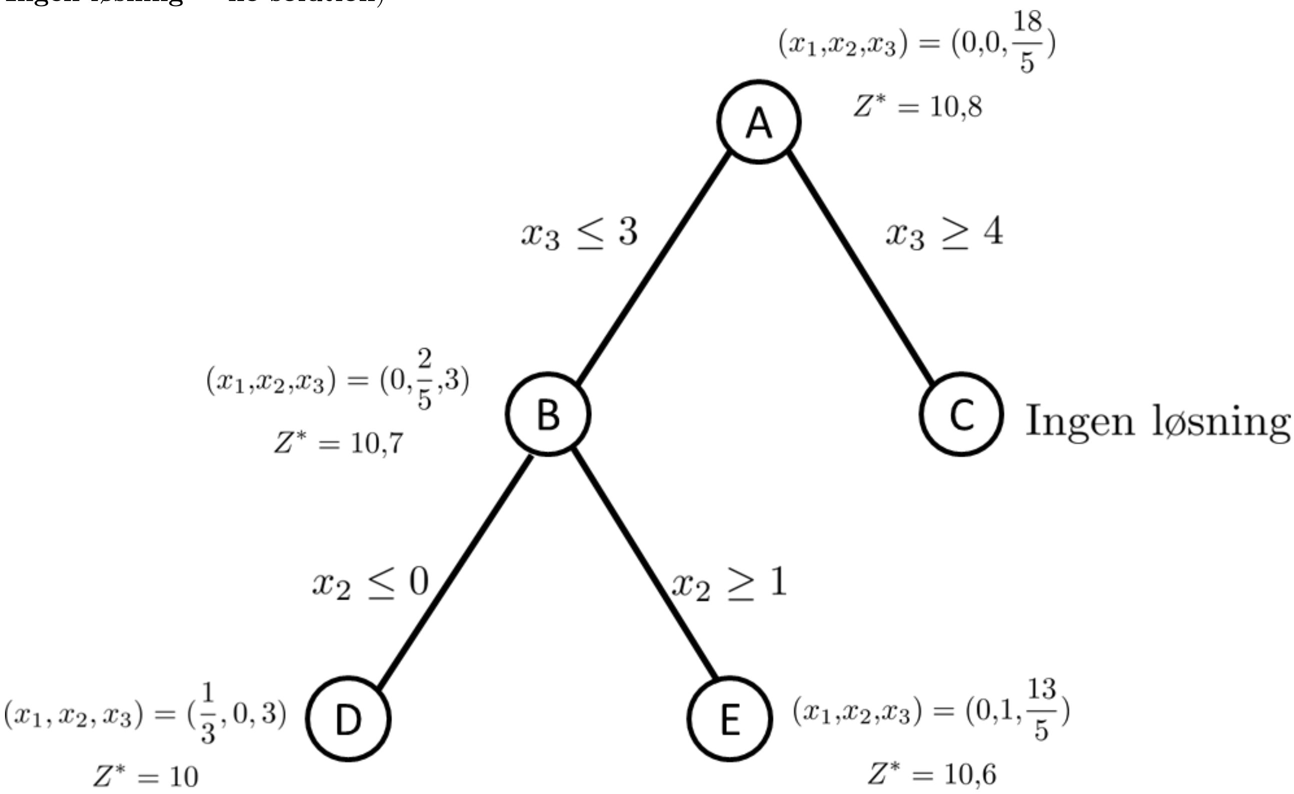
13F)

$$\max 30x_1 + 20x_2 - 3y$$

subject to

$$\begin{aligned} 0{,}7x_1 + 0{,}5x_2 &\leq 50 \\ 0{,}4x_1 + 0{,}25x_2 &\leq 20 \\ x_2 - y &\leq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

Exercise 14: Integer programming Your project group needs to solve a integer programming problem. The problem have three variables x_1, x_2 and x_3 which all are non negative integers. The objective function needs to be maximized and you know that $(x_1, x_2, x_3) = (0, 0, 1)$ fulfills all the constraints and has an objective value at 3. Your buddy started to solve the problem and has written the following branch-and-bound tree (**Note: Ingen løsning = no solution**)



At each knot the solution and objective value to the LP-relaxation is written. In knot C there was no feasible solutions to the LP-relaxation. You are not sure that your buddy completed the tree. Which of the following conclusions is most precise, given the information above?

14A) The optimal solution to the integer problem has an objective value in the interval 3 to 10.6.

14B) The optimal solution to the integer problem has an objective value at 3

14C) The optimal solution to the integer problem has an objective value at 10

14D) The optimal solution to the integer problem has an objective value at 10,6

14E) There is no feasible solution to the integer problem.

14F) The optimal solution to the integer problem has an objective value in the interval 3 to 10.7.

Exercise 15: Integer programming A company is expanding its product line and want to enter the football market. They are willing to produce 5 different products: footballs, shin pads and three kinds of football boots. Before they start the production they want to estimate whether the company will earn money by entering the marked.

Related to each product is a start up cost for buying machines that can produce the products. The start up cost and revenue for each sold product is given in the table below

Type	Shin pads	Football	Boot 1	Boot 2	Boot 3
Revenue	19	28	35	70	50
Start up cost	7000	25000	40000	25000	30000

Each of the products requires a certain amount of rubber and some kind of mesh material. One pair of shin pads requires 1 unit of rubber and 2 units of the mesh material. One football requires 2 units of rubber and 2 units of mesh material. Football boot 1 requires 3 units of rubber and 3 units of the mesh material, football boot 2 requires 5 units of rubber and 3 units of the mesh material and football boot 3 requires 6 units of rubber and 1 unit of the mesh material. The total amount of rubber available is 60000 and the total amount of mesh material available is 40000.
The company wants to have at least one type of football boot but they will not produce all three at a time. To be sure that they make more than just football boots they want to produce at least one other product meaning they have to produce either footballs, shin pads or both. The goal is to maximize the total revenue for the company based on the described situation. The assignment is to formulate the problem as a mixed integer problem and describe how you construct the constraints.

Exercise 16: Dual Simplex (text question, 10 points) Consider the following linear program:

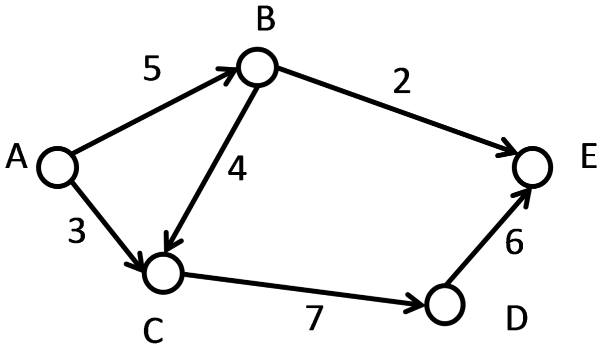
$$
\begin{aligned}
\texttt{Maximize:} \quad & Z = 3x_1 + 2x_2 + 5x_3 \\
\texttt{s.t.} \quad & x_1 + 2x_2 + x_3 + x_4 = 430 \\
& 3x_1 + 2x_3 + x_5 = 460 \\
& x_1 + 4x_2 + x_6 = 420 \\
& x_1, x_2, x_3, x_4, x_5, x_6 \geq 0
\end{aligned}
$$

The associated optimum tableau is as given as:

	Z	x_1	x_2	x_3	x_1	x_5	x_6	RHS
Z	1	4	0	0	1	2	0	1350
x_2	0	-$\frac{1}{4}$	1	0	$\frac{1}{2}$	-$\frac{1}{4}$	0	100
x_3	0	$\frac{3}{2}$	0	1	0	$\frac{1}{2}$	0	230
x_6	0	2	0	0	-2	1	1	20

You would like to now consider a new right hand side of 450 for Constraint 1, 460 for Constraint 2, and 400 for Constraint 3. Using Dual Simplex, and starting from the optimal primal tableau given above, find the new optimal solution. Write the optimal simplex tableau.

Exercise 17 (from spring 2015 exam) In this question we consider the network below. The arcs represent water pipes and the orientation of the arc determines the direction of water flow on the arc. A water source is present in vertex A and we wish to the send the water to vertex E. The numbers on the arcs indicate how many liters we can send per minute using each arc. The amount of water in A is considered unlimited so it is only the capacity of the arcs that determines how much water we can send to E per minute. We wish to write an LP that determines how many liters of water that can be send from A to E per minute. In the LP we use decision variables x_{ij} to represent how much water we send from vertex i to vertex j per minute. x_{BE}, for example, is the decision variable that tells how much water is sent from B to E, per minute.



Which of the following models provides the best representation of the problem?

17A)

$$\max x_{BE} + x_{DE}$$

subject to

$$\begin{aligned} x_{AB} - x_{BE} - x_{BC} &= 0 \\ x_{AC} + x_{BC} - x_{CD} &= 0 \\ x_{CD} - x_{DE} &= 0 \\ x_{AB} \leq 5, x_{AC} \leq 3, x_{BC} &\leq 4 \\ x_{BE} \leq 2, x_{CD} \leq 7, x_{DE} &\leq 6 \\ x_{AB}, x_{AC}, x_{BC}, x_{BE}, x_{CD}, x_{DE} &\geq 0 \end{aligned}$$

17B)

$$\min x_{BE} + x_{DE}$$

subject to

$$\begin{aligned} 5x_{AB} + 2x_{BE} - 4x_{BC} &= 0 \\ 3x_{AC} - 4x_{BC} - 7x_{CD} &= 0 \\ 7x_{CD} - 6x_{DE} &= 0 \\ x_{AB} \leq 5, x_{AC} \leq 3, x_{BC} &\leq 4 \\ x_{BE} \leq 2, x_{CD} \leq 7, x_{DE} &\leq 6 \\ x_{AB}, x_{AC}, x_{BC}, x_{BE}, x_{CD}, x_{DE} &\geq 0 \end{aligned}$$

17C)

$$\max x_{AB} + x_{AC} + x_{BC} + x_{BE} + x_{CD} + x_{DE}$$

subject to

$$\begin{aligned} x_{AB} + x_{BE} + x_{BC} &= 0 \\ x_{AC} - x_{BC} - x_{CD} &= 0 \\ x_{CD} - x_{DE} &= 0 \\ x_{AB} \leq 5, x_{AC} \leq 3, x_{BC} &\leq 4 \\ x_{BE} \leq 2, x_{CD} \leq 7, x_{DE} &\leq 6 \\ x_{AB}, x_{AC}, x_{BC}, x_{BE}, x_{CD}, x_{DE} &\geq 0 \end{aligned}$$

17D)

$$\max x_{BE} + x_{DE}$$

subject to

$$\begin{aligned} x_{AB} + x_{BE} + x_{BC} &= 11 \\ x_{AC} + x_{BC} + x_{CD} &= 14 \\ x_{CD} - x_{DE} &= 13 \\ x_{AB}, x_{AC}, x_{BC}, x_{BE}, x_{CD}, x_{DE} &\geq 0 \end{aligned}$$

17E)

$$\max 2x_{BE} + 6x_{DE}$$

subject to

$$\begin{aligned} x_{AB} - x_{BE} - x_{BC} &= 0 \\ x_{AC} + x_{BC} - x_{CD} &= 0 \\ x_{CD} - x_{DE} &= 0 \\ x_{AB}, x_{AC}, x_{BC}, x_{BE}, x_{CD}, x_{DE} &\geq 0 \end{aligned}$$

17F)

$$\max 5x_{AB} + 3x_{AC} + 4x_{BC} + 2x_{BE} + 7x_{CD} + 6x_{DE}$$

subject to

$$\begin{aligned} x_{AB} - x_{BE} - x_{BC} &= 0 \\ x_{AC} + x_{BC} - x_{CD} &= 0 \\ x_{CD} - x_{DE} &= 0 \\ x_{AB}, x_{AC}, x_{BC}, x_{BE}, x_{CD}, x_{DE} &\geq 0 \end{aligned}$$

15 Solutions exam F18

Exercise 1)

1C is the right answer.

$$\frac{2}{3}x_1 - x_2 \leq \frac{4}{3}$$
$$\frac{1}{2}x_1 + x_2 \leq 6$$
$$2x_1 + x_2 \leq 12$$
$$x_1, x_2 \geq 0$$

Exercise 2)

2C is the right answer.

$$(0,0) \rightarrow (0,6) \rightarrow (4,4)$$

Exercise 3

3F is the right answer.

b.v.	eq.	Z	x1	x2	x3	x4	x5	x6	RHS
Z	0	1	0	0	-2.5	4.5	0	-1.5	72
x2	1	0	0	1	0.5	0.5	0	-0.5	4
x5	2	0	0	0	3	0	1	-0.33	10
x1	3	0	1	0	0	0	0	0.33	4

Exercise 4

4D is the right answer.

b.v.	eq.	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	0	1	0	0	7	-1	0	10	58
x_1	1	0	1	0	1	-1	0	2	2
x_2	2	0	0	1	0	1	0	-1	5
x_5	3	0	0	0	-1	0	1	-2	1

Exercise 5

The right answer is 5D (use the SOB rule)
Primal:

$$\min 2x_1 + 6x_2 + 3x_3$$

$$x_1 + 4x_2 + 2x_3 = 6$$
$$2x_1 + 2x_2 + 3x_3 \leq 7$$
$$1x_1 + 2x_2 + 1x_3 \geq 1$$
$$x_1 \geq 0,\ x_2 \in \mathbb{R},\ x_3 \leq 0$$

Dual:

$$\max 6y_1 + 7y_2 + y_3$$

subject to

$$\begin{aligned} y_1 + 2y_2 + y_3 &\leq 2 \\ 4y_1 + 2y_2 + 2y_3 &= 6 \\ 2y_1 + 3y_2 + y_3 &\geq 3 \\ y_1 \in \mathbb{R}, y_2 \leq 0, y_3 &\geq 0 \end{aligned}$$

Exercise 6

Right answer:
6E) The dual problem to LP2 might not have a feasible solution. If there is a feasible solution the objective value is at least 42.

Explanation The optimal objective value for LP2, Z_2^*, will at least be 42 since the solution domain to LP2 includes all of the solutions to LP1. If LP2 is unconstrained there will be no solutions to the dual problem. If LP2 has a finite objective value it will be greater than 42 and the dual problem will have the same optimal objective value.

Exercise 7

Answer 7F is correct.
It is not necessary, but if we solve the original LP we get:

```
-------------------------------------------------------------------------
b.v.  | eq.  |     x1       x2       x3       x4       x5       x6  |   RHS
-------------------------------------------------------------------------
      |  0  |  -7.00    -8.00   -10.00     0.00     0.00     0.00  |   0.00
  x4  |  1  |   2.00     3.00     2.00     1.00     0.00     0.00  |   1.00
  x5  |  2  |   2.00     1.00     0.00     0.00     1.00     0.00  |  18.00
  x6  |  3  |   3.00     4.00     2.00     0.00     0.00     1.00  |   2.00
-------------------------------------------------------------------------
b.v.  | eq.  |     x1       x2       x3       x4       x5       x6  |   RHS
-------------------------------------------------------------------------
      |  0  |   3.00     7.00     0.00     5.00     0.00     0.00  |   5.00
  x3  |  1  |   1.00     1.50     1.00     0.50     0.00     0.00  |   0.50
  x5  |  2  |   2.00     1.00     0.00     0.00     1.00     0.00  |  18.00
  x6  |  3  |   1.00     1.00     0.00    -1.00     0.00     1.00  |   1.00
```

We use the information that x_3, x_5 og x_6 is basis variables to determine $B = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix}$, $c_B = \begin{bmatrix} 10 & 0 & 0 \end{bmatrix}$ og $B^{-1} = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$

The right hand side is

$$B^{-1}\left(b + \begin{bmatrix} 0 \\ 0 \\ \Delta \end{bmatrix}\right) = \begin{bmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}\left(b + \begin{bmatrix} 0 \\ 0 \\ \Delta \end{bmatrix}\right) = \begin{bmatrix} 0.5 \\ 18 \\ 1 + \Delta \end{bmatrix}$$

By doing sensitivity analysis we see that the basis remains legal when Δ is in the interval [-1;∞]. In this interval the solution is (as function of Δ):

```
Z=5.0
x3 = 0.5
```

We have $Z = 5$ (and not dependent on Δ) since the dual variable corresponding to the third constraint is zero (we can read that from the final tableau or compute using the fundamental insight).

Exercise 8

Answer 8B

b.v.	eq.	x1	x2	x3	x4	x5	x6	RHS	
	0	3	7	0	5	0	0	5	
		0	-1	1	0	0	0	0	Delta
x3	1	1	1.5	1	0.5	0	0	0.5	
x5	2	2	1	0	0	1	0	18	
x6	3	1	1	0	-1	0	1	1	
b.v.	eq.	x1	x2	x3	x4	x5	x6	RHS	
	0	3	7	0	5	0	0	5	
		-1	-2.5	0	-0.5	0	0	-0.5	Delta
x3	1	1	1.5	1	0.5	0	0	0.5	
x5	2	2	1	0	0	1	0	18	
x6	3	1	1	0	-1	0	1	1	

We construct the inequalities

$$3 - \Delta \geq 0 \Rightarrow \Delta \leq 3$$
$$7 - 2.5\Delta \geq 0 \Rightarrow 2.5\Delta \leq 7 \Rightarrow \Delta \leq 2.8$$
$$5 - 0.5\Delta \geq 0 \Rightarrow 0.5\Delta \leq 5 \Rightarrow \Delta \leq 10$$

So the basis is optimal when $\Delta \in [-\infty; 2.8]$ and the objective value in this interval will be $Z^* = 5 - 0,5\Delta$

Exercise 9

We first write the problem on augmented form. In order to do so we add a artificial variable $\bar{x}_4$ and a surplus variable x_5 for the first constraint, for the two next constraints we add a surplus variable (x_6 and x_7, respectively). The problem in augmented form is:

$$\max 7x_1 + 3x_2 + 1x_3$$

subject to

$$3x_1 + 2x_2 + \bar{x}_4 - x_5 = 2$$
$$4x_1 + 2x_2 + 5x_3 + x_6 = 7$$
$$1x_1 + 4x_2 + 2x_3 + x_7 = 14$$
$$x_1, x_2, x_3, \bar{x}_4, x_5, x_6, x_7 \geq 0$$

In phase 1 we aim at finding a feasible solution to the original problem and we therefore solve the problem with the objective

$$\min Z = \bar{x}_4$$

we rewrite to maximization to get

$$\max -Z = -\bar{x}_4$$

We write the problem on tableau form below: Tableau #0 is not on the proper form because of the “1“ under $\bar{x}_4$ in row 0. We subtract row 1 from row zero to obtain tableau #1. We do a single simplex iteration on tableau #1 to reach the optimal solution in tableau #2. We see that the only artificial variable x_4 has left the basis and we have found a feasible solution to the original problem

#	b.v.	eq.	Z	x1	x2	x3	x4	x5	x6	x7	RHS
0	Z	0	-1	0	0	0	1	0	0	0	0
	x4	1	0	3	2	0	1	-1	0	0	2
	x6	2	0	4	2	5	0	0	1	0	7
	x7	3	0	1	4	2	0	0	0	1	14
1	Z	0	-1	-3	-2	0	0	1	0	0	-2
	x4	1	0	3	2	0	1	-1	0	0	2
	x6	2	0	4	2	5	0	0	1	0	7
	x7	3	0	1	4	2	0	0	0	1	14
2	Z	0	-1	0	0	0	1	0	0	0	0
	x1	1	0	1	$\frac{2}{3}$	0	$\frac{1}{3}$	$-\frac{1}{3}$	0	0	$\frac{2}{3}$
	x6	2	0	0	$-\frac{2}{3}$	5	$-\frac{4}{3}$	$\frac{4}{3}$	1	0	$\frac{13}{3}$
	x7	3	0	0	$\frac{10}{3}$	2	$-\frac{1}{3}$	$\frac{1}{3}$	0	1	$\frac{40}{3}$

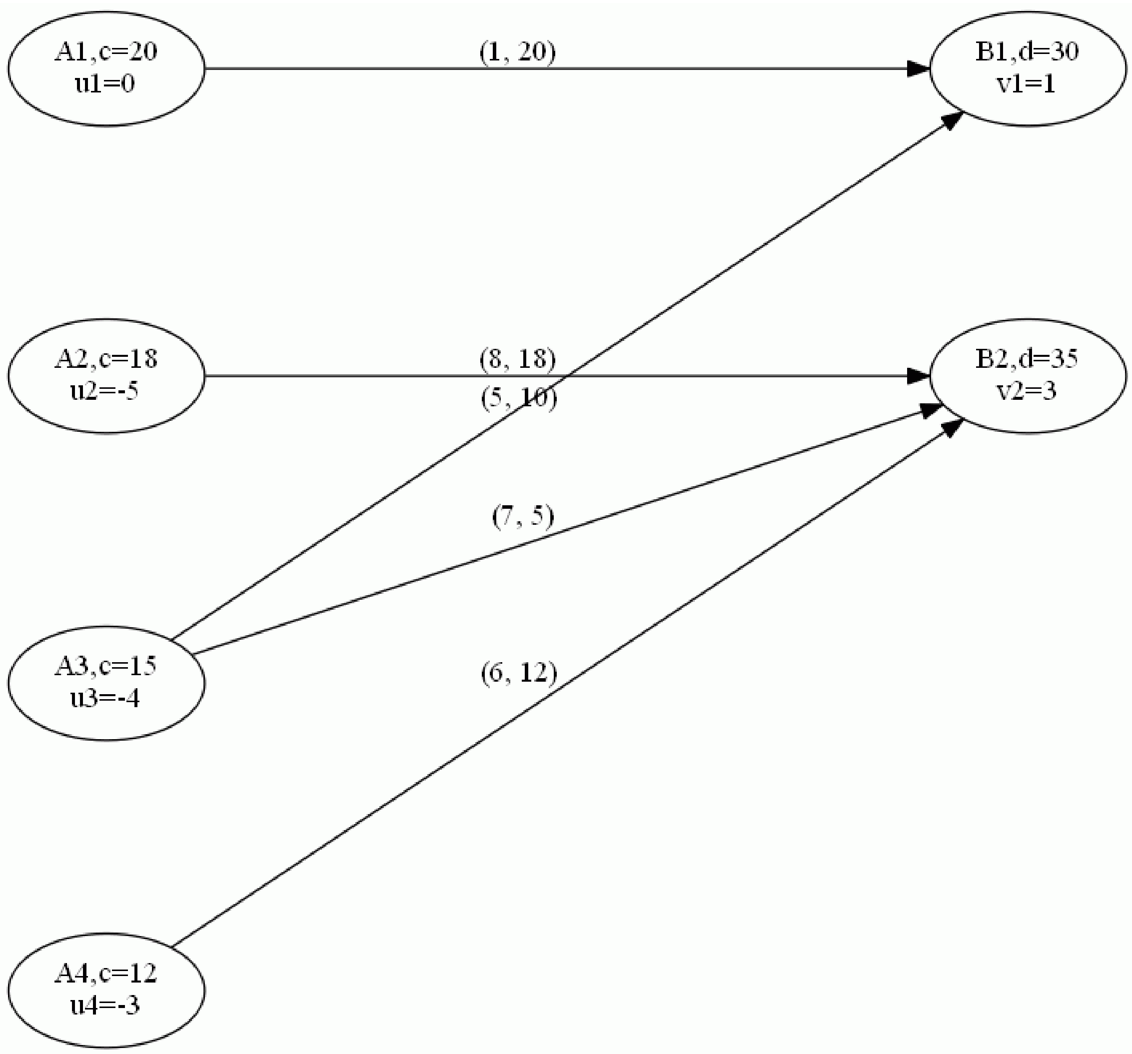
We are now ready to proceed to phase 2. We erase the artificial variable $\bar{x}_4$ and insert the original objective. The result of this is shown tableau #3 below. This tableau is not on the proper form since x_1 is a basis variable and it has a non-zero coefficient in row 0. We add 7 times row 1 to row zero to get tableau #4. In tableau #4 we see that x_5 should enter and $x_6$should leave the basis. We perform the pivot operation and ends with tableau #5 that is optimal.

#	b.v.	eq.	Z	x1	x2	x3	x4	x5	x6	x7	RHS
3	Z	0	1	-7	-3	-1		0	0	0	0
	x1	1	0	1	$\frac{2}{3}$	0		$-\frac{1}{3}$	0	0	$\frac{2}{3}$
	x6	2	0	0	$-\frac{2}{3}$	5		$\frac{4}{3}$	1	0	$\frac{13}{3}$
	x7	3	0	0	$\frac{10}{3}$	2		$\frac{1}{3}$	0	1	$\frac{40}{3}$
4	Z	0	1	0	$\frac{5}{3}$	-1		$-\frac{7}{3}$	0	0	$\frac{14}{3}$
	x1	1	0	1	$\frac{2}{3}$	0		$-\frac{1}{3}$	0	0	$\frac{2}{3}$
	x6	2	0	0	$-\frac{2}{3}$	5		$\frac{4}{3}$	1	0	$\frac{13}{3}$
	x7	3	0	0	$\frac{10}{3}$	2		$\frac{1}{3}$	0	1	$\frac{40}{3}$
5	Z	0	1	0	0.5	7.75		0	1.75	0	12.25
	x1	1	0	1	0.5	1.25		0	0.25	0	1.75
	x5	2	0	0	-0.5	3.75		1	0.75	0	3.25
	x7	3	0	0	3.5	0.75		0	-0.25	1	12.25

We see that the optimal solution is to set $(x_1, x_2, x_3) = (1.75, 0, 0)$ which result in the objective value $Z^* = 12.25$

Exercise 10

10C

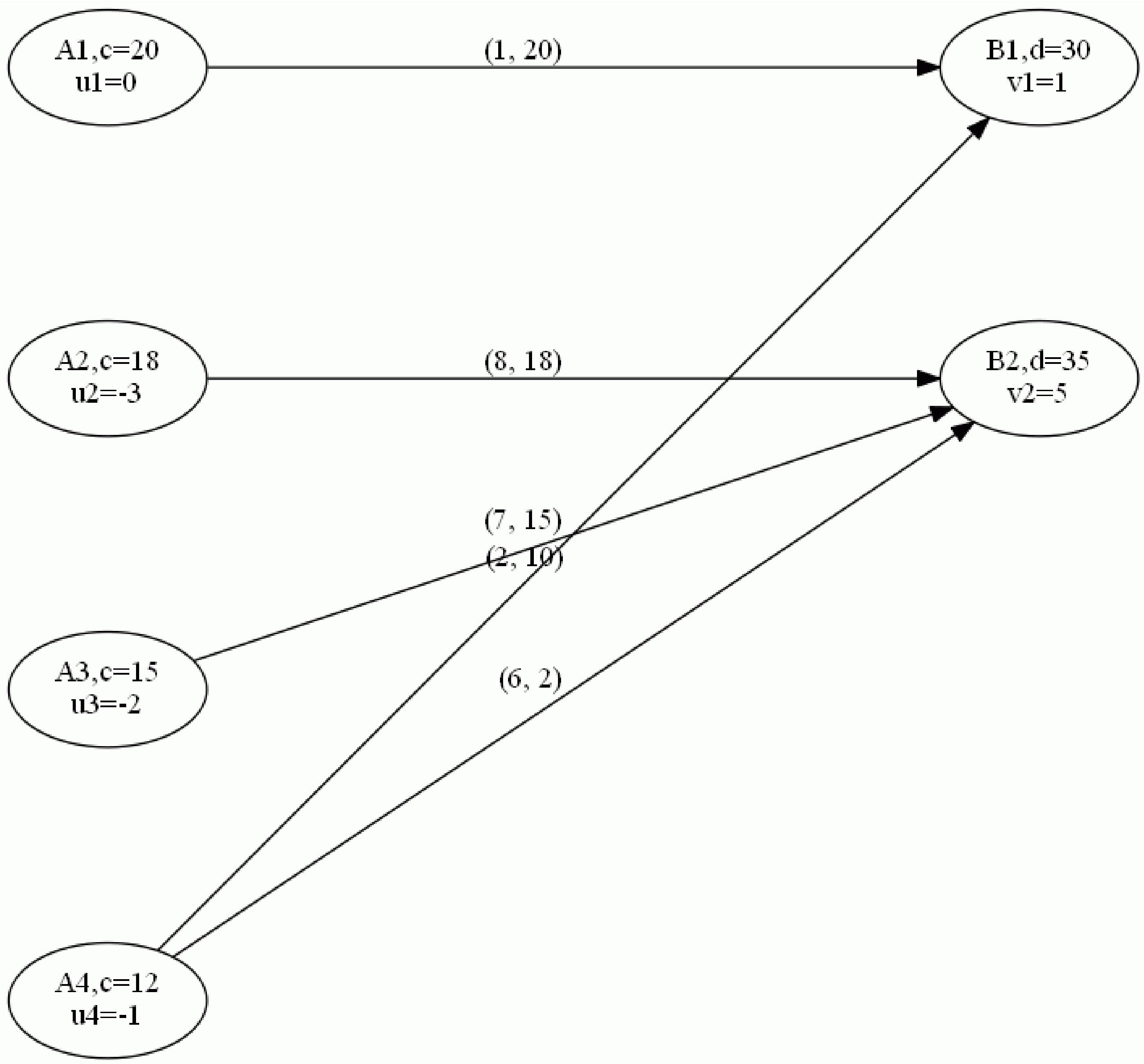



```
red cost of arc (A1,B2): -1
red cost of arc (A2,B1): 3
red cost of arc (A4,B1): -2
```

Exercise 11

Answer 11E is correct
The edge (A4,B1) enters basis since it has the most negative reduced cost. A4, B1, A3, B2, A4 is now fully connected and we have flow moving backwards from (A3,B1) and (A4,B2). There are 10 units flowing backward at the first egde and 12 at the second. Hence, (A3,B1) leaves basis and (A4,B1) will get a flow at 10 units.

Fra	A1	A2	A3	A4	A4
Til	B1	B2	B2	B1	B2
Antal	20	18	15	10	2



Exercise 12

Missing answer

opg 13

Answer 13F is correct

$$\max 30x_1 + 20x_2 - 3y$$

subject to

$$\begin{aligned} 0,7x_1 + 0,5x_2 &\leq 50 \\ 0,4x_1 + 0,25x_2 &\leq 20 \\ x_2 - y &\leq 15 \\ x_1, x_2, y &\in \mathbb{Z}_+ \end{aligned}$$

Exercise 14

Answer: 14A
The optimal solution to the integer problem has an objective value in the interval 3 to 10.6.

Exercise 15

Introduce a integer variable x_i and a binary variable y_i for each product. The integer variable represent the production amount of product i and the binary variable represent if the production of product i is started such that when $y_i = 1$, product i is being produced. Product 1 is the shin pads, product 2 is the football, product 3 is boot 1, product 4 is boot 2 and product 5 is boot 3.
The objective function is to maximize the total profit given the revenue and the start up cost:

$$\max Z = 19x_1 + 28x_2 + 35x_3 + 70x_4 + 50x_5 - 7000y_1 - 25000y_2 - 40000y_3 - 25000y_4 - 30000y_5$$

The requirement of at least one type of football boot and less than three type of football boots can be expressed in two constraints:

$$\begin{aligned} y_3 + y_4 + y_5 &\geq 1 \\ y_3 + y_4 + y_5 &\leq 2 \end{aligned}$$

It is required that at least one of the two product types: shin pads or football is being produced. This can be expressed as:

$$y_1 + y_2 \geq 1 \tag{102}$$

The limitations of rubber and mesh material can be described with two constraint. The first constraint represent the limitation of rubber and the second constraint represent the limitation of mesh material.

$$\begin{aligned} 1x_1 + 2x_2 + 3x_3 + 5x_4 + 6x_5 &\leq 60000 \\ 2x_1 + 2x_2 + 3x_3 + 3x_4 + 1x_5 &\leq 40000 \end{aligned}$$

Lastly we need to make sure that product i only can be produced when $y_i = 1$.

$$x_i \leq My_i \qquad \forall i = 1, 2, 3, 4, 5$$

where M is a big value.
The complete formulation is:

$$\begin{aligned} \max Z = 19x_1 + 28x_2 + 35x_3 + 70x_4 + 50x_5 & \\ - 7000y_1 - 25000y_2 - 40000y_3 - 25000y_4 - 30000y_5 & \\ y_3 + y_4 + y_5 &\geq 1 \\ y_3 + y_4 + y_5 &\leq 2 \\ y_1 + y_2 &\geq 1 \\ 1x_1 + 2x_2 + 3x_3 + 5x_4 + 6x_5 &\leq 60000 \\ 2x_1 + 2x_2 + 3x_3 + 3x_4 + 1x_5 &\leq 40000 \\ x_i &\leq My_i \qquad \forall i = 1, 2, 3, 4, 5 \\ x_i &\in \mathbb{Z} \qquad \forall i = 1, 2, 3, 4, 5 \\ y_i &\in \{0, 1\} \qquad \forall i = 1, 2, 3, 4, 5 \end{aligned}$$

Exercise 16

Revised optimal tableau with new right handside and optimal tableau after one dual simplex pivot

	Z	x_1	x_2	x_3	x_4	x_5	x_6	RHS
Z	1	4	0	0	1	2	0	1370
x_2	0	$-\frac{1}{4}$	1	0	$\frac{1}{2}$	$-\frac{1}{4}$	0	110
x_3	0	$\frac{3}{2}$	0	1	0	$\frac{1}{2}$	0	230
x_6	0	2	0	0	-2	1	1	-40

	Z	x_1	x_2	x_3	x_4	x_5	x_6	
Z	1	5	0	0	1	$\frac{5}{2}$	$\frac{1}{2}$	1350
x_2	0	$\frac{1}{4}$	1	0	0	0	$\frac{1}{4}$	100
x_3	0	$\frac{3}{2}$	0	1	0	$\frac{1}{2}$	0	230
x_4	0	-1	0	0	1	$-\frac{1}{2}$	$-\frac{1}{2}$	20

Exercise 17

Correct answer: **17A**

Reason: The objective function maximizes the amount of water that flows into vertex E. The first constraint ensures that the water that flows into B continues to vertex C or E. Constraint 2 and 3 works in the same way, but for vertex C and D. Upper bounds on the x variables ensure that we respect the capacity limits. Lower bounds on the x variables ensure that we do not “cheat” and send water in the opposite direction of the arc.