Constructor University
Fall 2023

November 16, 2023

Operations Research

Midterm Exam (Make-up)

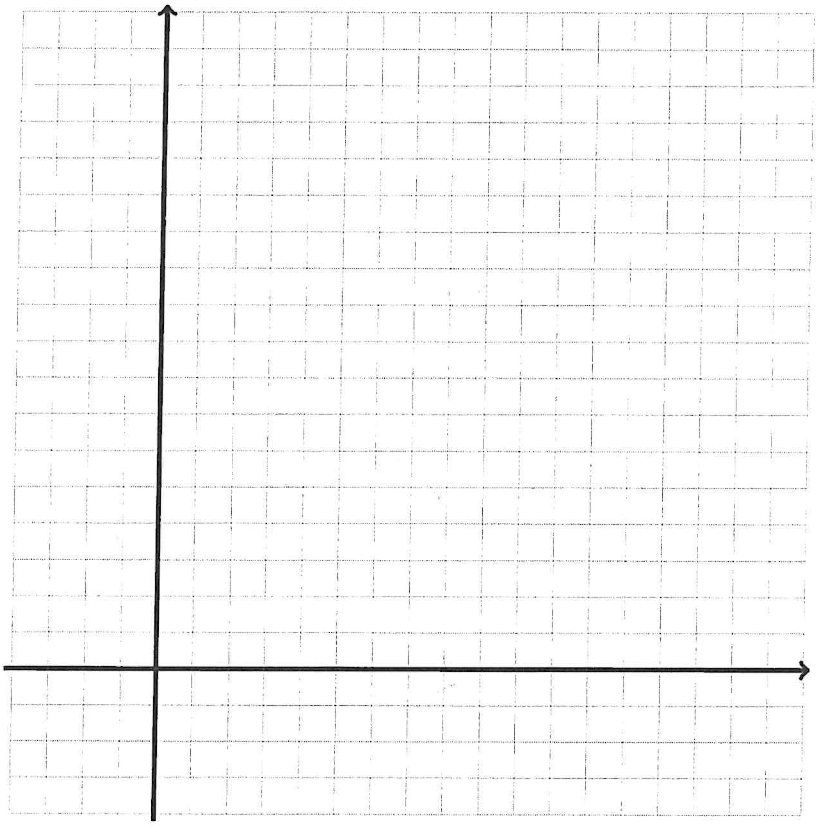
(For bonus points only.)

Instructions:

- Do all the work on this exam paper.
- Show your work, i.e., carefully write down the steps of your solution. You will receive points not just based on your final answer, but on the correct steps in your solution.
- No tools or other resources are allowed for this exam. In particular, no notes and no calculators.

Name: Solutions

Matric. No.: ____________



Problem 1: Graphical Method [25 points]

Consider the following Linear Programming (LP) problem LP1: Maximize

$$Z = x_1 + x_2$$

subject to

$$\frac{1}{2}x_1 - x_2 \leq 1,$$
$$3x_1 + x_2 \leq 7,$$
$$x_1, x_2 \geq 0.$$

(6) (a) Solve the problem with the graphical method only, i.e., draw the feasible region, and read off the optimal x_1, x_2 (as far as the accuracy of the picture permits), and the optimal Z.

(5) (b) From part (a), identify the two binding constraints. Using this information, compute the optimal solution and the corresponding value of Z exactly.

(8) (c) Now consider another Linear Programming problem, LP2: Minimize

$$\widetilde{Z} = y_1 + 7y_2$$

subject to

$$\frac{1}{2}y_1 + 3y_2 \geq 1,$$
$$-y_1 + y_2 \geq 1,$$
$$y_1, y_2 \geq 0.$$

Solve this LP problem as well with the graphical method, and compute the optimal y_1, y_2 and $\widetilde{Z}$.

(3) (d) What is the meaning and interpretation of problem LP2 from (c) compared to the original LP problem LP1 above?

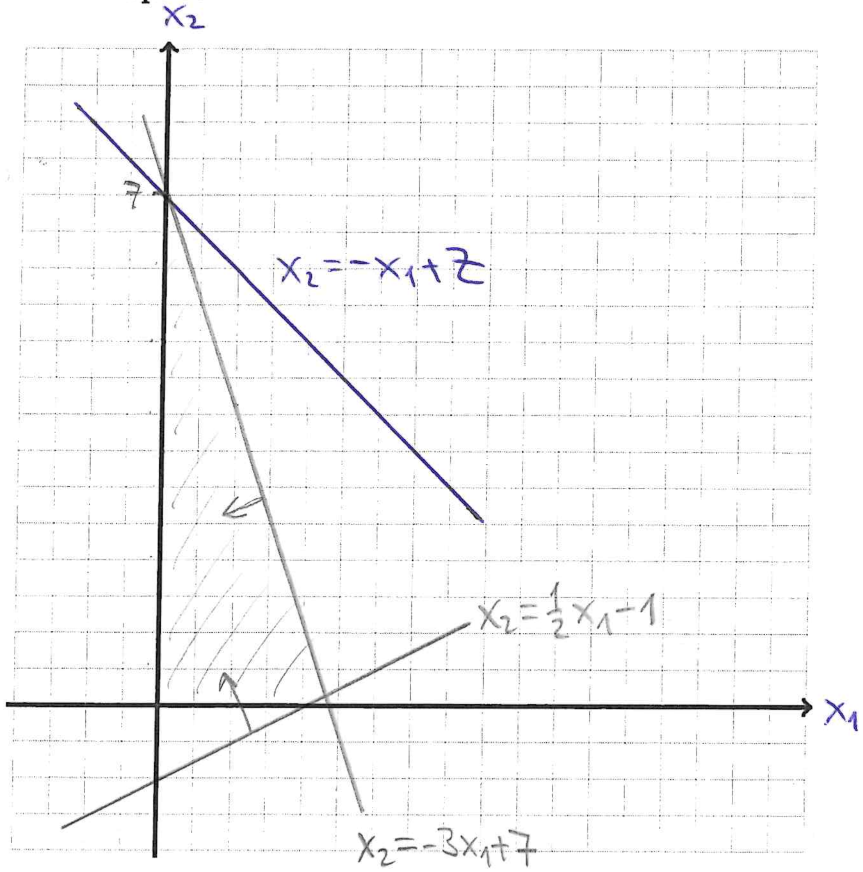
(3) (e) Consider again LP1 above. *Using only parts (c) and (d)*, determine the new optimal Z if:

(i) we increase the right-hand side of the first constraint from 1 to 2;

(ii) we decrease the right-hand side of the second constraint from 7 to 6.

Problem 1: Extra Space

(6) a)



optimal solution is $x_1 = 0, x_2 = 7$ with $z = 7$.

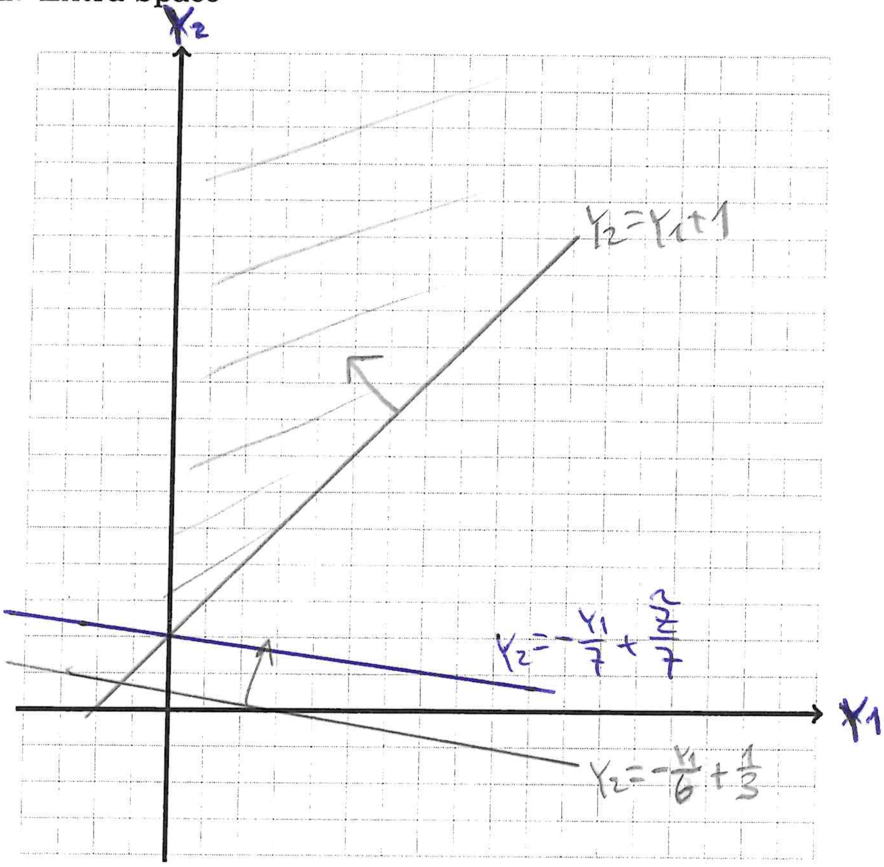
(7) b) Binding constraints: $x_1 \geq 0$ and $3x_1 + x_2 \leq 7$

Optimal sol. is at the intersection of $x_1 = 0$ and $3x_1 + x_2 = 7$

$\Rightarrow x_1 = 0, x_2 = 7$, with $z = 0 + 7 = 7$

Problem 1: Extra Space

(8) c)



Binding constraints are $y_1 \geq 0$ and $-y_1 + y_2 \geq 1$.

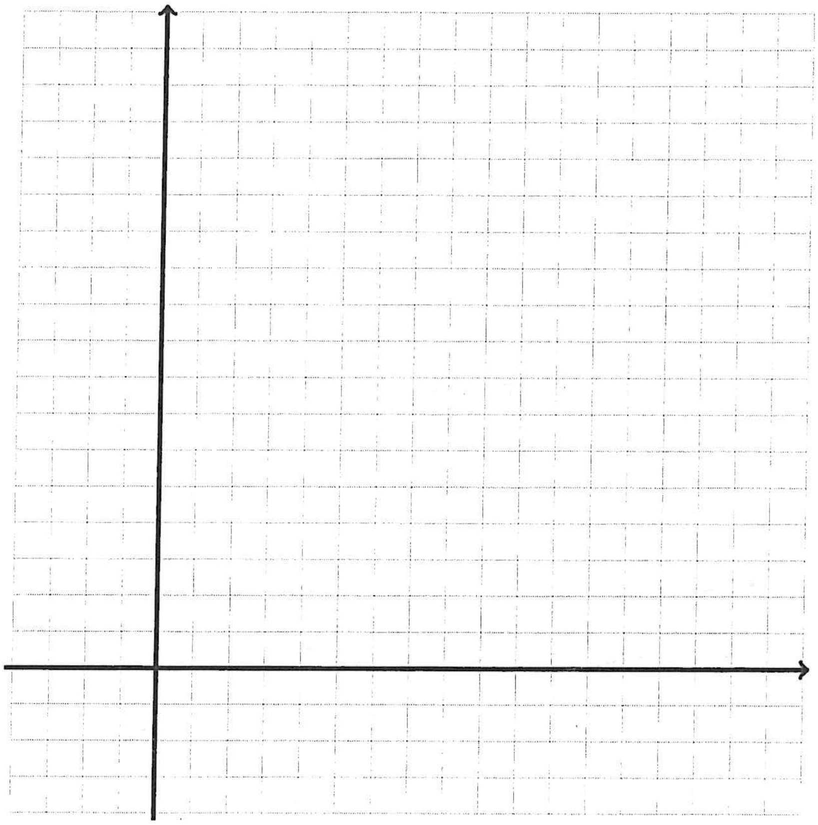
$\Rightarrow$ optimal sol. is $y_1 = 0$, $y_2 = 1$, with $Z = 7$.

d) LP2 is the dual problem to LP1. Solving the dual gives us the shadow prices y_1, y_2, which tell us how Z changes if we change the capacities in the constraints by a small amount.

e) (i) Increasing from 1 to 2 does not change Z, since $y_1 = 0$.

(ii) Decreasing from 7 to 6 changes Z by $(-1) \cdot y_2^{opt} = -1$, i.e. the new Z is 6.

Problem 1: Extra Space



Problem 2: Standard Form and Simplex Method [25 points]

(10) (a) Consider the following Linear Programming problem: Maximize

$$Z = 3x_1 + 2x_2 + 5x_3$$

subject to

$$\begin{aligned} x_1 + 3x_2 &\leq 7, \\ 2x_1 - 5x_2 - 3x_3 &\geq 1, \\ x_2, x_3 &\geq 0 \quad \text{(no nonnegativity constraint on } x_1\text{)}. \end{aligned}$$

Write this problem in standard form, i.e., as the problem to minimize

$$\tilde{Z} = c^{\mathrm{T}} \tilde{x}$$

subject to

$$A\tilde{x} = b, \quad \tilde{x} \geq 0.$$

Specify exactly the vectors b, c and the matrix A.

(15) (b) Consider the following Linear Programming problem: Maximize

$$Z = \frac{1}{3}x_1 + x_2$$

subject to

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

Solve the problem with the simplex method as shown in class. *(Note: You will not receive points for simply stating the solution, but only for using the simplex method step by step.)*

Problem 2: Extra Space

(10) a) Step 1: Minimize $-z = -3x_1 - 2x_2 - 5x_3$

Step 2: slack variables

- first constraint: $x_1 + 3x_2 + s_1 = 7 \quad , s_1 \geq 0$
- second constraint: $-2x_1 + 5x_2 + 3x_3 \leq -1$

$$\Rightarrow -2x_1 + 5x_2 + 3x_3 + s_2 = -1 \quad , s_2 \geq 0$$

Step 3: x_1 has no negativity constraint, so we introduce

$$x_1 = u - v \quad \text{with } u, v \geq 0$$

$$\Rightarrow \tilde{x} = \begin{pmatrix} u \\ v \\ x_2 \\ x_3 \\ s_1 \\ s_2 \end{pmatrix} , \quad A = \begin{array}{c} \begin{array}{cccccc} u & v & x_2 & x_3 & s_1 & s_2 \end{array} \\ \begin{pmatrix} 1 & -1 & 3 & 0 & 1 & 0 \\ -2 & 2 & 5 & 3 & 0 & 1 \end{pmatrix} \end{array}$$

$$b = \begin{pmatrix} 7 \\ -1 \end{pmatrix} , \quad c = \begin{pmatrix} -3 \\ 3 \\ -2 \\ -5 \\ 0 \\ 0 \end{pmatrix}$$

Problem 2: Extra Space

(15) b) Simplex tableau:

$$
\begin{array}{cccc|c}
1 & -1 & 1 & 0 & 1 \\
-\frac{1}{3} & 1 & 0 & 1 & 3 \\
\hline
-\frac{1}{3} & -1 & 0 & 0 & 0
\end{array}
$$

$$
\begin{array}{l|cccc|c}
R_2+R_1 & \frac{2}{3} & 0 & 1 & 1 & 4 \\
 & -\frac{1}{3} & 1 & 0 & 1 & 3 \\
\hline
R_2+R_3 & -\frac{2}{3} & 0 & 0 & 1 & 3
\end{array}
$$

$$
\begin{array}{l|cccc|c}
\frac{3}{2}R_1 & 1 & 0 & \frac{3}{2} & \frac{3}{2} & 6 \\
\frac{1}{2}R_1+R_2 & 0 & 1 & \frac{1}{2} & \frac{3}{2} & 5 \\
\hline
R_1+R_3 & 0 & 0 & 1 & 2 & 7
\end{array}
$$

optimal solution is $x_1=6, x_2=5$ with $z=7$

Problem 2: Extra Space

Problem 3: Network optimization [25 points]
A company has 4 factories that produce goods that need to be delivered to 3 warehouses. The available supply s_i at factory i, the demand d_j at warehouse j, and the shipping costs c_{ij} to ship from factory i to warehouse j are as in the following table:

	Warehouse			
	1	2	3	Supply
Factory 1	\$300	\$400	\$300	20
Factory 2	\$200	\$800	\$500	20
Factory 3	\$300	\$500	\$700	10
Factory 4	\$400	\$400	\$600	10
Demand	14	16	30	

We are looking for the most cost effective way to ship all the supply from the factories to the warehouses.

(5) (a) Draw a network representation of the problem.

(10) (b) Formulate this transportation problem as a Linear Programming problem, i.e., introduce appropriate decision variables, write down the objective function (and in which sense it is to be optimized) and the constraints.

(5) (c) Are there feasible solutions to this LP problem for the data given in the table? Briefly explain your answer.

(5) (d) Determine which of the following is the optimal solution to this LP problem. (Note that (i, j) means the amount shipped from factory i to warehouse j.) Clearly explain why you exclude certain possibilities!

```
Solution 1:
{(1, 1): 0.0,
 (1, 2): 0.0,
 (1, 3): 20.0,
 (2, 1): 10.0,
 (2, 2): 0.0,
 (2, 3): 10.0,
 (3, 1): 4.0,
 (3, 2): 6.0,
 (3, 3): 0.0,
 (4, 1): 0.0,
 (4, 2): 10.0,
 (4, 3): 0.0}
```

Could be the optimal sol.

```
Solution 2:
{(1, 1): 0.0,
 (1, 2): 0.0,
 (1, 3): 20.0,
 (2, 1): 10.0,
 (2, 2): 0.0,
 (2, 3): 0.0,
 (3, 1): 4.0,
 (3, 2): 6.0,
 (3, 3): 0.0,
 (4, 1): 10.0,
 (4, 2): 10.0,
 (4, 3): 0.0}
```

Cannot be optimal, e.g. because from Factory 2 only 10 units are shipped, but the supply is 20.

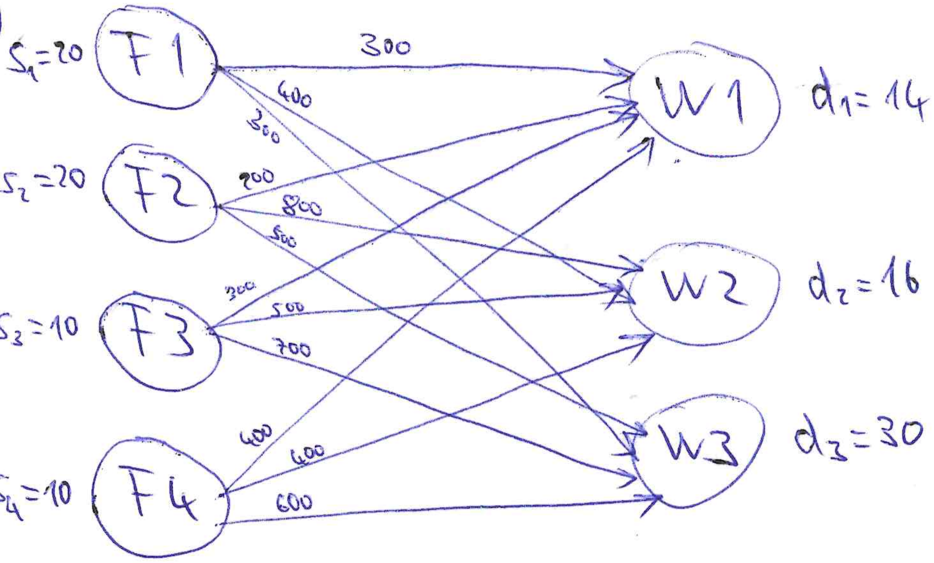
```
Solution 3:
{(1, 1): 0.0,
 (1, 2): 0.5,
 (1, 3): 19.5,
 (2, 1): 10.0,
 (2, 2): 0.0,
 (2, 3): 10.0,
 (3, 1): 4.5,
 (3, 2): 5.5,
 (3, 3): 0.0,
 (4, 1): 0.5,
 (4, 2): 9.5,
 (4, 3): 0.0}
```

Cannot be optimal because in the optimal sol. all x_{ij} must be integer

Problem 3: Extra Space

c_{ij} as in table

a)



b) Decision variables: x_{ij} = # of units shipped from factory i to warehouse j, $i=1,2,3,4$, $j=1,2,3$

Objective: Minimize cost $Z = \sum_{i=1}^{4} \sum_{j=1}^{3} c_{ij} x_{ij}$

subject to $\sum_{j=1}^{3} x_{ij} = s_i \quad \forall i = 1,2,3,4$

$\sum_{i=1}^{4} x_{ij} = d_j \quad \forall j = 1,2,3$

$x_{ij} \geq 0 \quad \forall i,j$

c) Yes, there is always an optimal sol. if supply = demand.
Here, total supply $= 20+20+10+10 = 60$ and total demand $= 14+16+30 = 60$.

Problem 3: Extra Space

Problem 3: Extra Space

Problem 4: Pyomo [25 points]

The pyomo program on the last exam page shows an implementation of a minimum cost flow problem (which we discussed in class).

(6) (a) Draw the corresponding network representation of the problem. (Make sure to include the given values of the parameters from the pyomo program.)

(5) (b) Write down the Linear Programming problem that is solved in mathematical notation (i.e., write down the objective function and the constraints).

(5) (c) Briefly explain the meaning of the parameters b, C, U in the context of minimum cost flow problems. In particular, explain what positive, negative, or zero values for entries in b mean.

(3) (d) Suppose one of the entries in U can be slightly increased. Which one should we choose in order to lower the costs? (You will receive points only if you justify your answer correctly, and not for guessing.)

(6) (e) Now consider the following situation. For cost saving reasons the distribution center has to be closed down, as well as the connections between F1 and F2, and between W1 and W2. In exchange, there are now transportation connections between F1 and W2 (with cost $C_{F1,W2} = 500$), F2 and W1 (with cost $C_{F1,W2} = 400$), and F2 and W2 (with cost $C_{F1,W2} = 400$), in addition to the existing connection from F1 to W1. There are no more capacity constraints, and the values in b stay the same. (In other words, the problem turned into a transportation problem.) Modify the code such that it would solve this new problem.

a) - d) See Fall 2022 midterm make-up exam solution

e) See code at the end.

Problem 4: Extra Space

Problem 4: Extra Space

Problem 4: Extra Space

```
from pyomo.environ import *
from pyomo.opt import *
opt = solvers.SolverFactory("glpk")
```

```
b = {'F1':50,
     'F2':40,
     'DC':0,
     'W1':-30,
     'W2':-60}

C = {('F1','F2'):200,
     ('F1','DC'):400,
     ('F1','W1'):900,
     ('F2','DC'):300,
     ('DC','W2'):100,
     ('W1','W2'):300,
     ('W2','W1'):200}

U = {('F1','F2'):10,
     ('DC','W2'):80}

N = list(b.keys())
A = list(C.keys())
V = list(U.keys())
```

new: C = {('F1','W1'): 900,
('F1','W2'): 500,
('F2','W1'): 400,
('F2','W2'): 400}

```
model = ConcreteModel()
model.x = Var(A, within=NonNegativeReals)
```

```
def flow_rule(model, n):
    InFlow  = sum(model.x[i,j] for (i,j) in A if j==n)
    OutFlow = sum(model.x[i,j] for (i,j) in A if i==n)
    return OutFlow - InFlow == b[n]

model.flow = Constraint(N, rule=flow_rule)
```

```
def capacity_rule(model, i, j):
    return model.x[i,j] <= U[i,j]

model.capacity = Constraint(V, rule=capacity_rule)
```

```
model.cost = Objective(expr = sum(model.x[a]*C[a] for a in A), sense=minimize)
```

```
model.dual = Suffix(direction=Suffix.IMPORT)
results = opt.solve(model)
model.x.get_values()
```

```
{('F1', 'F2'): 0.0,
 ('F1', 'DC'): 40.0,
 ('F1', 'W1'): 10.0,
 ('F2', 'DC'): 40.0,
 ('DC', 'W2'): 80.0,
 ('W1', 'W2'): 0.0,
 ('W2', 'W1'): 20.0}
```

```
model.cost.expr()
```

```
49000.0
```

```
for n in N:
    print(n, model.dual[model.flow[n]])
```

```
F1 700.0
F2 600.0
DC 300.0
W1 -200.0
W2 0.0
```

```
for j in V:
    print(j, model.dual[model.capacity[j]])

('F1', 'F2') 0.0
('DC', 'W2') -200.0
```