Exam Operations Research

Date: March 29, 2019
Time: 12:00 - 14:45

Points per exercise:

- Exercise 1 has a total of 25 points (a(10), b(5) and c(10)).
- Exercise 2 has a total of 15 points.
- Exercise 3 has a total of 20 points (a(5), b(5) and c(10)).
- Exercise 4 has a total of 15 points (a(10) and b(5)).
- Exercise 5 has a total of 15 points (a(5) and b(10)).

Thus in total 90 points can be obtained. The exam grade is determined as follows:

$$\text{Exam grade } = 1 + \frac{\text{total number of obtained points}}{10}.$$

$$\text{Final course grade} = \frac{1}{4}\text{Weekly pretest grade} + \frac{3}{4}\text{Exam grade.}$$

- Non-programmable, non-graphing calculators are allowed.
- This exam consists of 4 pages, including this one.
- The duration of this exam is **2 hours and 45 minutes**.
  Students who have obtained permission for extra time may use an *additional 30 minutes*. This is only possible in the room allocated for this purpose.

Exercise 1

Consider the following LP which is referred to as the "primal LP".

$$\begin{aligned} \max \quad & z = -2x_1 + x_2 + x_3 + x_4 \\ \text{s.t.} \quad & -x_1 + 3x_2 + 2x_3 + 2x_4 \leq 30 \\ & x_1 + 2x_2 + 3x_3 + x_4 = 20 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{aligned}$$

(a) **[10 points]** Determine the dual of this LP.

(b) **[5 points]** The primal LP has been solved by the simplex method resulting in the following final tableau:

Basic	z	x_1	x_2	x_3	x_4	s_1	r_1	value
z	1	1.50	0.50	0	0	0.50	0	15.00
x_4	0	−1.25	1.25	0	1	0.75	−0.50	12.50
x_3	0	0.75	0.25	1	0	−0.25	0.50	2.50

Determine the optimal solution of the primal LP and optimal objective value. Determine also the optimal solution of the dual LP and optimal dual objective function.

(c) **[10 points]** The right hand side value 30 of the first constraint of the primal LP is replaced by the value 50. Apply the dual simplex method to determine the optimal solution and optimal objective value of the modified LP.

Solution exercise 1:

(a) The dual LP is:

$$\begin{aligned}
\min \quad & w = 30y_1 + 20y_2 \\
\text{s.t.} \quad & -y_1 + y_2 \geq -2 \\
& 3y_1 + 2y_2 \geq 1 \\
& 2y_1 + 3y_2 \geq 1 \\
& y_1 \geq 0, y_2 \text{ unrestricted}
\end{aligned}$$

(b) The optimal solution of the primal LP is $x_1^* = 0$, $x_2^* = 0$, $x_3^* = 2.50$, $x_4^* = 12.50$ and optimal objective value $z^* = 15.00$. The optimal solution of the dual LP is $y_1^* = 0.50$, $y_2^* = 0$ with optimal dual objective value $w^* = 15.00$.

(c) Since $\Delta_1 = 50 - 30 = 20$ the modified simplex tableau is:

Basic	z	x_1	x_2	x_3	x_4	s_1	r_1	value
z	1	1.50	0.50	0	0	0.50	0	25.00
x_4	0	-1.25	1.25	0	1	0.75	-0.50	27.50
x_3	0	0.75	0.25	1	0	-0.25	0.50	-2.50

Now apply one iteration of the dual simplex method with x_3 leaving the basis and s_1 entering the basis. Then we obtain the following tableau:

Basic	z	x_1	x_2	x_3	x_4	s_1	r_1	value
z	1	3	1	2	0	0	1	20.00
x_4	0	1	2	3	1	0	1	20.00
s_1	0	-3	-1	-4	0	1	-2	10.00

Hence the optimal solution of the modified LP is $x_1^* = x_2^* = x_3^* = 0$, $x_4^* = 20$ with optimal objective value $z^* = 20.00$.

Exercise 2 [15 points]

A product has to be transported from a factory to a distribution center to fulfill the demand to this product in the upcoming 5 weeks. Let d_i be the demand in week i for $i = 1, 2, \ldots, 5$. To transport the product trucks of different types j, $j = 1, 2, 3$ can be used. A type j truck can transport at most b_j units of the product per transport. The cost per transport with a type j truck is c_j. It is not possible to have in week i more than a_{ij} transports with a type j truck.

The demand in each week i has to be fulfilled, but it is allowed to transport more units than what is needed to fulfill the demand. In that case the units exceeding the demand will be stored at the distribution center to fulfill demand in future weeks. The storage costs are h per unit per week.

Formulate the problem of minimizing the total (of transport and storage) costs such that the demand to the product is fulfilled for each week i and satisfying all other restrictions as an integer linear program. Explain the meaning of all variables and constraints you have in your integer linear program formulation.

Solution exercise 2:

Let x_{ij} be the number of type j trucks used in week i. Let y_i be the number of units transported in week

i. Let z_i the number of units to be stored at the end of week i. Then we obtain the following ILP formulation:

$$\begin{aligned}
\min \quad & w = \sum_{i=1}^{5}\sum_{j=1}^{3} c_j x_{ij} + h\sum_{i=1}^{5} z_i \\
\text{s.t.} \quad & y_1 - z_1 = d_1 \\
& y_i - z_i + z_{i-1} = d_i && \text{for } i = 2,3,4,5 \\
& y_i \le \sum_{j=1}^{3} b_j x_{ij} && \text{for } i = 1,2,3,4,5 \\
& x_{ij} \le a_{ij} && \text{for all } i,j \\
& \text{All variables } x_{ij}, y_i, z_i \ge 0, \quad \text{all variables } x_{ij}, y_i, z_i \text{ integer}
\end{aligned}$$

Exercise 3

Consider the following problem. You are packing your suitcase for a journey and need to decide what items to bring. You have 4 items (just one of each); each has a certain value to you, and a certain weight:

weight (kg)	2	6	7	3
value	6	19	22	10

You do not want to have more than 13 kg of weight in the suitcase. The problem is to choose under that weight restriction which items should be put into the suitcase maximizing the total value of those items.

(a) **[5 points]** Formulate an integer linear program for this problem.

(b) **[5 points]** Determine the unique optimal solution of the LP relaxation of this problem. Also determine the corresponding optimal objective value.

(c) **[10 points]** Solve the ILP from (a) by applying the **branch-and-bound** method. Clearly indicate in which order you compute the nodes of the search tree and where you prune the search tree. Also indicate which pruning criterion you use when you prune.

Solution exercise 3:

(a)

$$\begin{aligned}
\max \quad & z = 6x_1 + 19x_2 + 22x_3 + 10x_4 \\
\text{s.t.} \quad & 2x_1 + 6x_2 + 7x_3 + 3x_4 \le 13 \\
& x_1, x_2, x_3, x_4 \in \{0, 1\}
\end{aligned}$$

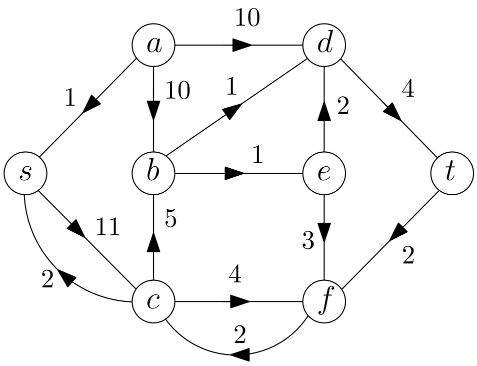
(b) Since $\frac{10}{3} > \frac{19}{6} > \frac{22}{7} > \frac{6}{2}$ it follows for this binary knapsack problem that the LP relaxation has optimal solution $x_1 = 0$, $x_2 = 1$, $x_3 = \frac{4}{7}$, $x_4 = 1$ with $z^* = 41\frac{4}{7}$.

(c) From (b) we already have the upperbound $z^0 = 41$ since the value is always integer for feasible solutions. We branch on variable x_3.
Put $x_3 = 0$ for subproblem 1. Then the LP relaxation has optimal solution $x_1 = x_2 = 1, x_3 = 0, x_4 = 1$ yielding upper bound $z^1 = 35$ which is also a lower bound for the original problem since this solution is feasible. We do not have to branch further from subproblem 1. Put $x_3 = 1$ for subproblem 2. Then the LP relaxation has optimal solution $x_1 = 0$, $x_2 = \frac{3}{6} = \frac{1}{2}$, $x_3 = 1$, $x_4 = 1$ yielding upper bound $z^2 = 41\frac{1}{2}$. Branch on variable x_2 from this node.
Put $x_2 = 0, x_3 = 1$ for subproblem 2.1. Then the LP relaxation has optimal solution $x_1 = 1$, $x_2 = 0$,

$x_3 = 1$, $x_4 = 1$ yielding upper bound $z^{2.1} = 38$ which is also a new best lower bound for the original problem since this solution is feasible. We do not have to branch further from subproblem 2.1.
Put $x_2 = 1, x_3 = 1$ for subproblem 2.2. Then the LP relaxation has optimal solution $x_1 = 0$, $x_2 = 1$, $x_3 = 1$, $x_4 = 0$ yielding upper bound $z^{2.2} = 41$ which is also a new best lower bound for the original problem since this solution is feasible. We do not have to branch further from subproblem 2.2.
We conclude that the original problem is solved now. The optimal solution is $x_1 = 0$, $x_2 = 1$, $x_3 = 1$, $x_4 = 0$ with optimal value $z^* = 41$.

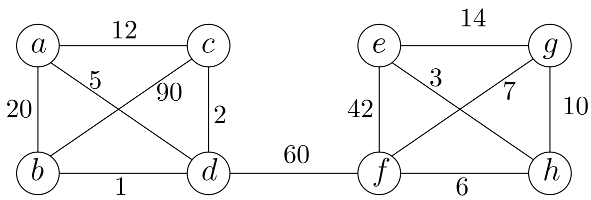
Exercise 4

(a) **[10 points]** Consider the instance of the maximum flow problem shown below (arc capacities are indicated). Determine a maximum flow from s to t using the Ford-Fulkerson algorithm; show your working, and be clear about how you know that the maximum flow has been found.



(b) **[5 points]** Consider the problem of finding a minimum weight spanning tree in the graph below using Kruskal's algorithm, where edge weights have been indicated.

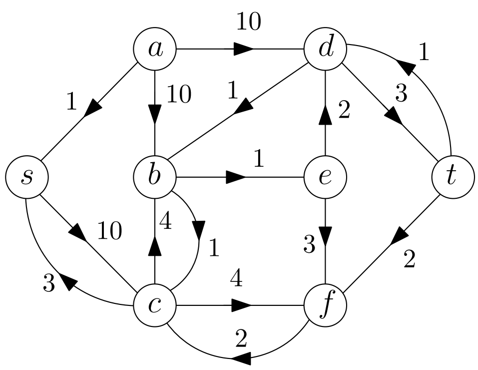
Without running the whole algorithm, state i) which edge Kruskal's algorithm will pick first, and ii) which edge will be the final edge picked by the algorithm. Explain your answers briefly.

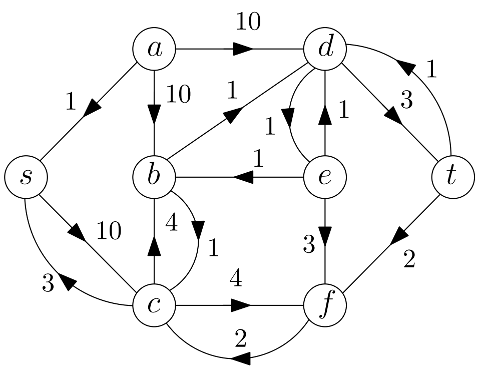


Solution exercise 4:

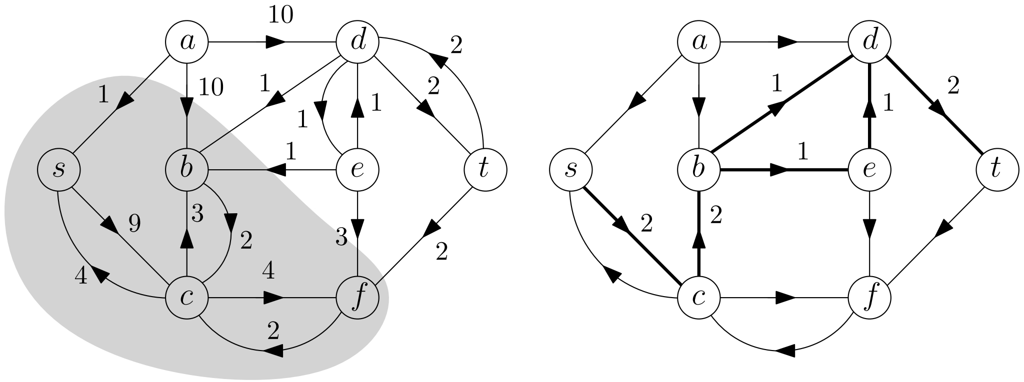
(a) The first augmenting path can be either s, c, b, d, t or s, c, b, e, d, t. In either case we augment by 1 unit, since the smallest capacity arc is 1.

The residual graph after this first augmentation is shown below (the left being for the first choice of path).





In the left case, the next augmentation will be s, c, b, e, d, t, and in the right case, s, c, b, d, t. After this, the same residual graph will be obtained, and there is no longer a path from s to t. More precisely, there are no arcs in the residual graph leaving $S = \{s, c, b, f\}$. So the flow is now optimal. The image below shows the final residual graph, as well as the actual optimal flow on the right.



(b) i) Edge $\{b, d\}$ will be picked first, since it has the smallest weight.

ii) Edge $\{d, f\}$ will be picked last. The only edge of larger weight is $\{b, c\}$ but this will not be picked, because $\{c, d\}$ will be picked second, and so $\{b, d\}, \{c, d\}, \{d, f\}$ forms a cycle.

Exercise 5

Recall the knapsack problem. We are given an integer B representing the size of the knapsack, and n item types labelled 1 through n, with item type i having integer size $a_i \in \mathbb{N}$ and cost $c_i \geq 0$. The goal is to determine how many items of each type to put in the knapsack, so that the items fit and we obtain the largest possible value.

(a) **[5 points]** Consider the following instance of the knapsack problem.

$$B = 10, a_1 = 3, a_2 = 4, a_3 = 5, c_1 = 5, c_2 = 7, c_3 = 8.$$

Determine, **using dynamic programming**, the optimal solution to this problem. Determine not just the objective value, but the actual solution.

Instruction: To obtain any points, it must be clear from what you write down that you have used dynamic programming to solve the problem.

(b) **[10 points]** Now consider the following variation of the knapsack problem. In addition to the requirement that the items fit, we must pick at most N items in total (where N is a given positive integer).

Give a dynamic programming algorithm for the problem, by filling in the missing parts ([[1]], [[2]], [[3]]) of the template below. You must decide what the meaning of the subproblems $K(b, r)$ is: explain your choice.

```
Let a = min{a_1, a_2, ..., a_n}
Initialization: [[1]]
for b = a, a+1, ..., B do
    for r = 1, 2, ..., N do
        K(b, r) = [[2]]
return [[3]]
```

Solution exercise 5:

(a)

b	1	2	3	4	5	6	7	8	9	10
profit	0	0	5	**7**	8	10	**12**	14	15	**17**

An optimal solution is: buy twice item type 1, and once item type 2. (The bold items show a possible "path" back from 10).

(b)
- [[1]]: $K(b, r) = 0$ if $b < a$, or if $r = 0$.
- [[2]]: $K(b, r) = \max_{i:a_i \leq b}\big(K(b - a_i, r - 1) + c_i\big)$
- [[3]]: return $K(B, N)$.

The meaning of $K(b, r)$ is the maximum profit attainable in a knapsack of size b with at most r items.