You are requested to provide a report in pdf format answering problems 5.1 and 5.2 and a notebook for problem 5.3.

Please submit your answers to this page before the given deadline. Late submissions will be penalized by 5 points per day after the deadline.

Problem 5.1: IP formulations

Consider the set $X = \{(0,0,0,0),(1,0,0,0),(0,1,0,0),(0,0,1,0),(0,0,0,1),(0,1,0,1),(0,0,1,1)\}$.

(a) Verify that the following three polyhedra are formulations of X.

$$P_1 = \{\mathbf{x} \in \mathbb{R}^4_+ : 0 \leq x \leq 1,\ 83x_1 + 61x_2 + 49x_3 + 20x_4 \leq 100\}$$
$$P_2 = \{\mathbf{x} \in \mathbb{R}^4_+ : 0 \leq x \leq 1,\ 4x_1 + 3x_2 + 2x_3 + 1x_4 \leq 4\}$$
$$P_3 = \{\mathbf{x} \in \mathbb{R}^4_+ : 0 \leq x \leq 1,\ 4x_1 + 3x_2 + 2x_3 + 1x_4 \leq 4,\ x_1 + x_2 + x_3 \leq 1,\ x_1 + x_4 \leq 1\}$$

(b) Which of the three formulations P_1, P_2, or P_3 is the best one for X and why?

Problem 5.2: B&B formulation for knapsack

(a) Consider the following 0-1 Knapsack problem with $N = \{1, \dots, n\}$ items:

$$\text{(0-1 KP)} \quad \max_x. \left\{ \sum_{i=1}^n c_i x_i : \sum_{i=1}^n a_i x_i \leq b, x \in \{0,1\}^n \right\},$$

with item weights $a_i > 0$ and item profits $c_i > 0$ for all items $i \in N$. If $\sum_{i=1}^n a_i \leq b$, the capacity constraint cannot be violated and we have a trivial solution where $x_i = 1$ for all $i \in N$. Show that if $c_1/a_1 \geq \dots \geq c_n/a_n$, $\sum_{i=1}^{r-1} a_i \leq b$, and $\sum_{i=1}^{r} a_i > b$ for some index $r \leq n$, then the optimal solution to the linear programming (LP) relaxation of the problem 0-1 KP is $x_i = 1$ for all $i \in N$ such that $i < r$, $x_r = (b - \sum_{i=1}^{r-1} a_i)/a_r$, and $x_i = 0$ for all $i \in N$ such that $i > r$. The LP-relaxation of 0-1 KP is otherwise identical to the 0-1 KP formulation but $x \in \{0,1\}^n$ is relaxed to $0 \leq x_i \leq 1$ for all $i \in N$.

Hint: You can try to prove this by writing the dual problem of the LP-relaxation, finding a dual feasible solution, and verifying optimality by complementary slackness conditions (as done in Exercise 5.5). However, you can also give a proof "sketch" (i.e., it does not have to be a formal proof) that justifies why the proposed solution is optimal to the LP relaxation.

(b) Solve the following 0-1 Knapsack problem instance using LP-relaxation based B&B.

$$\begin{aligned} \text{(0-1 KP)} \quad \max_x. \quad & 17x_1 + 10x_2 + 25x_3 + 17x_4 \\ \text{s.t.:} \quad & 5x_1 + 3x_2 + 8x_3 + 7x_4 \leq 12 \\ & x_i \in \{0,1\}, \quad i = 1, \dots, 4. \end{aligned}$$

Solve the LP relaxations at each node of the B&B tree by using the result of part (a). Clearly show the structure of the B&B tree, and justify your reasoning when pruning nodes in the tree. Start the B&B procedure without any primal bound.

Problem 5.3: GAP formulation

Consider the following variation of the Generalized Assignment Problem (GAP). We have $M = \{1, \dots, m\}$ jobs and $N = \{1, \dots, n\}$ agents. Each agent $j \in N$ has a maximum work capacity b_j. If a job $i \in M$ is assigned to an agent $j \in N$, it generates a profit c_{ij}, but consumes a_{ij} units of agent j's work capacity b_j. The problem is to find a maximum profit assignment of jobs to agents so that:

(1) Each job $i \in M$ is assigned to at most one agent $j \in N$.
(2) Total capacity of jobs assigned to an agent $j \in N$ does not exceed b_j.

The problem can be formulated as follows.

$$
\begin{aligned}
\text{(GAP)} \quad \max_{x}. \quad & \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} \\
\text{s.t.} \quad & \sum_{j \in N} x_{ij} \leq 1, && \forall i \in M, \\
& \sum_{i \in M} a_{ij} x_{ij} \leq b_j, && \forall j \in N, \\
& x_{ij} \in \{0, 1\}, && \forall i \in M, \forall j \in N,
\end{aligned}
$$

where $x_{ij} = 1$ if job $i \in M$ is assigned to agent $j \in N$, and $x_{ij} = 0$ otherwise.

Consider an instance with $M = \{1, \ldots, 4\}$ jobs and $N = \{1, 2\}$ agents. The profits (c_{ij}) and job capacities (a_{ij}) for all $i \in M, j \in N$ are

$$
(c_{ij}) = \begin{pmatrix} 6 & 2 \\ 7 & 7 \\ 4 & 8 \\ 3 & 3 \end{pmatrix}, \quad (a_{ij}) = \begin{pmatrix} 5 & 3 \\ 7 & 1 \\ 4 & 6 \\ 2 & 4 \end{pmatrix},
$$

and the agents' work capacities are $b_1 = 7$ and $b_2 = 6$. Implement (the model) and solve the problem with Julia using JuMP. Return your Julia code in notebook format (.ipynb). The implementation should be fairly straightforward, and we ask you to focus also on the readability of your code. Submit a notebook that is easy to understand for someone who knows the problem formulation and some Julia. Three points will be given for the correct implementation and two for a *reasonably* well structured and commented notebook for a total of five points.

A few generic tips for improving the readability of your code are:

- Create as many cells as needed to separate different parts of your answer. A good example for that is leaving the first cells for calling the libraries needed, separate cells for plots or function definitions, etc.
- Write enough comments
- Make clear what is the purpose of each and every variable/object created along the code
- Separate the data and the model. If we change the instance (matrices a, b and c) to 50 jobs and 20 agents, the same JuMP model should still work
- Present the results clearly