Question 1 For each of (P1), (P2), (P3), (P4), (P5) and (P6) indicate **LP** in the box if it is a linear programming problem, **IP** if it is an integer programming problem, or **None** if it is neither. No justification necessary.

$$\begin{aligned} \max \quad & \sum_{j=1}^{n} c_j x_j \\ \text{s.t.} \quad & \sum_{j=1}^{n} A_{ij} x_j \le \sin(b_i), \quad \forall i = 1, \dots, m \\ & x \in \mathbb{Z}^n, \end{aligned} \tag{P1}$$

where $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$.

Problem (P1) is of type:

$$\begin{aligned} \max \quad & x_1 + x_2 \\ \text{s.t.} \quad & Ax = b \\ & x \ge 0 \\ & x \in \mathbb{R}^n, \end{aligned} \tag{P2}$$

where $A \in \mathbb{Z}^{m \times n}$ and $b \in \mathbb{Z}^m$.

Problem (P2) is of type:

$$\begin{aligned} \max \quad & x_2 \\ \text{s.t.} \quad & Gx \le b \\ & a_1 x_1 + a_2 x_2 \le 1, \quad \forall a \in \mathbb{R}^2 \; : \; ||a|| = 1 \\ & x \in \mathbb{R}^n, \end{aligned} \tag{P3}$$

where $G \in \mathbb{Z}^{m \times n}$ and $b \in \mathbb{Z}^m$.

Problem (P3) is of type:

$$\begin{aligned} \max \quad & x_2 \\ \text{s.t.} \quad & Gx \le b \\ & a_j x_j \le 1, \quad \forall j = 1, \dots, n, \; \forall a_j \in \mathbb{R} \; : \; |a_j| = 1 \\ & x \in \mathbb{R}^n, \end{aligned} \tag{P4}$$

where $G \in \mathbb{Z}^{m \times n}$ and $b \in \mathbb{Z}^m$.

Problem (P4) is of type:

$$\begin{aligned} \max \quad & c^T x \\ \text{s.t.} \quad & GAx \le b \\ & x \in \mathbb{R}^n, \end{aligned} \tag{P5}$$

where $b, c \in \mathbb{R}^n$ and $A, G \in \mathbb{Z}^{n \times n}$.

Problem (P5) is of type:

$$
\begin{aligned}
\max \quad & \log(x_1^2+1) \\
\text{s.t.} \quad & Ax \le b \\
& x \in \mathbb{Z}^n,
\end{aligned}
\tag{P6}
$$

where $b \in \mathbb{Z}^m$, $c \in \mathbb{R}^n$, $A \in \mathbb{Z}^{m\times n}$.

Problem (P6) is of type: $\boxed{}$

Question 2

(a) Let us define m functions $f(x) : \mathbb{R}^n \to \mathbb{R}$ for $i = 1, \ldots, m$, given by

$$f^i(x) := \max\{x + i, ix + 1\}.$$

Consider the problem

$$
\begin{aligned}
\min \quad & c^T x \\
\text{s.t.} \quad & f^i(x) \le 1, \quad \forall i = 1, \ldots, m
\end{aligned}
\tag{M1}
$$

where $c \in \mathbb{R}^m$. Formulate (M1) as an LP.

(b) Consider the problem

$$
\begin{aligned}
\max \quad & \cos(-x_1) \\
\text{s.t.} \quad & Ax = b \\
& 0 \le x \le 2 \\
& x \in \mathbb{R}^n,
\end{aligned}
\tag{M2}
$$

where $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$. Formulate (M2) as an LP.

Question 3.1 We are scheduling E exams for S students. The exam period has D days and each day has 4 exam time slots. We are given a matrix $A \in \mathbb{R}^{S\times E}$ such that for every student $s \in \{1, \ldots, S\}$ and every exam $e \in \{1, \ldots, E\}$, $A_{se} = 1$ if student s needs to take exam e, and $A_{se} = 0$ otherwise. A student cannot take two exams during the same time slot of the same day. An exam time of a given student is called *problematic* if the student has another exams less than three days before (i.e., we want two non-exam days between exams). We allow at most one *problematic* exam per student. Model the problem of choosing the exam dates and time slots to minimize the number of students having a *problematic* exam.

Question 3.2 We are given a list of 40 people who are to participate in a team exercise. The 40 participants must be split into 5 teams of 8. Each participant i has a known skill $s_i \in [0, 1]$ at the exercise. The skill of a team is the sum of the skills of the participants. Formulate as an integer programming model the problem of forming teams while minimizing the difference between the highest team skill and the lowest team skill.

Question 3.3 We are playing a simplified version of the Battleship game. At the beginning of the game, two players each have a 10×10 grid in which to place non-overlapping ships. In this simplified version of the game, we only place 4 identical cruisers (ships that take 3 consecutive tiles on the grid), and we can only place them vertically. Players do not see the ship arrangement of their opponent. Then, the two players take turns taking one shot at their opponent's grid, by indicating the shot coordinates (i, j). Every time, the other player only responds by indicating whether a ship has been hit or sunk (all ship tiles were hit). The game ends when all of one player's ships are sunk.

The position of the ships should be fixed throughout the game, but we are cheating. At each turn, after the opponent has called a shot $(\hat{\imath}, \hat{\jmath})$, but before replying whether a boat was hit or not, we move the ships in a way that is consistent with the previous shots but attempts to avoid further damage. To decide how to move the ships, we solve an integer programming problem.

The data we have is given by two matrices $H, M \in \mathbb{R}^{10\times 10}$. $H_{ij} = 1$ if a previous shot at coordinates (i, j) hit a ship, otherwise $H_{ij} = 0$. $M_{ij} = 1$ if a previous shot at coordinates (i, j) was a miss, otherwise $H_{ij} = 0$. The objective function models that fact that we attempt to avoid a hit at $(\hat{\imath}, \hat{\jmath})$. Write the integer programming model that we need to cheat at this game. (Generalization: model the IP necessary to cheat at the real game of Battleship.)

Question 3.4 We are planifying the weekly production of a steel coil manufacturing facility for the upcoming year. The order book is full, so the weekly demand d_w is known for the whole period $w = 1, \ldots, 52$. On any given week w, the facility is either *stopped* or *running*. Once the facility is started (i.e. it switches from *stopped* to *running*), it needs to be kept *running* for at least 3 consecutive weeks. When the facility is *running*, production has a known varying cost $c_w > 0$ per coil on week w, plus a fixed cost $f > 0$ per week. In addition, coils can be stored at a cost $s > 0$ per coil per week. Before the first week, the facility is not producing, and there are no coils in storage. When the facility is *stopped*, we only pay for storage. Before the beginning of the year, the facility is *stopped* and has no coils in storage. Devise an integer programming model to find a production plan that minimizes the costs.

Question 4.1 Consider the linear programming problem

$$\begin{array}{rrcrcrcrcl} \min & 2x_1 & + & 1x_2 & + & 3x_3 & + & 3x_4 & & \\ \text{s.t.} & 1x_1 & + & 2x_2 & - & 1x_3 & + & 2x_4 & = & 4 \\ & 1x_1 & - & 1x_2 & - & 2x_3 & + & 2x_4 & = & 2 \\ & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq & 0. \end{array} \tag{S1}$$

An optimal basis is $\mathcal{B} = \{2, 4\}$.

(a) Compute the tableau, basis matrix B and its inverse B^{-1} associated with the basis $\mathcal{B}$.

(b) Compute the allowable range for all right-hand sides.

(c) Compute the allowable range for all costs.

(d) If the right-hand side of the second constraint is changed from 2 to -3, apply the dual simplex method to compute a new optimal tableau.

Question 4.2 Consider the linear programming problem

$$\begin{array}{rrcrcrcrcl} \min & 0x_1 & + & 1x_2 & - & 1x_3 & + & 2x_4 & & \\ \text{s.t.} & 2x_1 & + & 1x_2 & - & 1x_3 & + & 1x_4 & = & 4 \\ & -1x_1 & + & 2x_2 & - & 2x_3 & - & 1x_4 & = & 4 \\ & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq & 0. \end{array} \tag{S1}$$

An optimal basis is $\mathcal{B} = \{1, 2\}$.

(a) Compute the tableau, basis matrix B and its inverse B^{-1} associated with the basis $\mathcal{B}$.

(b) Compute the allowable range for all right-hand sides.

(c) Compute the allowable range for all costs.

(d) If the right-hand side of the second constraint is changed from 4 to -3, apply the dual simplex method to compute a new optimal tableau.

Question 5.1 Consider the following problem:

$$\begin{array}{rrcrcrcrcl} \max & 6x_1 & + & 6x_2 & + & 3x_3 & + & 4x_4 & & \\ \text{s.t.} & 9x_1 & + & 3x_2 & + & 1x_3 & + & 3x_4 & \leq & 6 \\ & & & & x \in \{0,1\}^4. & & & & & \end{array} \tag{K1}$$

Solve (K1) using the branch-and-bound method. At any given node of the branch-and-bound tree, if the optimal LP solution is $\tilde{x}$ with $\tilde{x}_i \notin \mathbb{Z}$ and branching is required, always fully explore the subtree in which $x_i = 0$ before exploring the subtree in which $x_i = 1$. (In technical terms, perform a depth-first search on the branch-and-bound tree, always starting with the "$= 0$" branch.) For every node, give the optimal LP solution and its objective function value (or write "infeasible"), and if branching is not required, specify why (pruning, infeasible, integer). Draw the branch-and-bound tree.

Question 5.2 Consider the following problem:

$$\begin{array}{rrcrcrcrcl} \max & 2x_1 & + & 3x_2 & + & 6x_3 & + & 1x_4 & & \\ \text{s.t.} & 8x_1 & + & 2x_2 & + & 5x_3 & + & 4x_4 & \leq & 6 \\ & & & & x \in \{0,1\}^4. & & & & & \end{array} \tag{K2}$$

Solve (K2) using the branch-and-bound method. At any given node of the branch-and-bound tree: (i) give the optimal LP solution and its objective function value, or write "infeasible", (ii) if branching is required, always fully explore the subtree in which $x_i = 0$ before exploring the subtree in which $x_i = 1$, and (iii) and if branching is not required, specify why. Draw the branch-and-bound tree.

Question 5.3 Consider the following problem:

$$\begin{array}{rrcrcrcrcl} \max & 2x_1 & + & 5x_2 & + & 6x_3 & + & 1x_4 & & \\ \text{s.t.} & 3x_1 & + & 10x_2 & + & 2x_3 & + & 5x_4 & \leq & 14 \\ & & & & x \in \{0,1\}^4. & & & & & \end{array} \tag{K3}$$

Solve (K3) using the branch-and-bound method. At any given node of the branch-and-bound tree: (i) give the optimal LP solution and its objective function value, or write "infeasible", (ii) if branching is required, always fully explore the subtree in which $x_i = 0$ before exploring the subtree in which $x_i = 1$, and (iii) and if branching is not required, specify why. Draw the branch-and-bound tree.

Question 6 Let $A \in \mathbb{Z}^{m \times n}$ be a $-1, 0, +1$ matrix in which each column has at most one $+1$ entry and at most one -1 entry. Prove that A is TU.

Question 7.1 Put the following Boolean formula in conjunctive normal form (CNF).

$$(\ ((\neg x_3) \vee x_4)\ \wedge\ (\neg(x_2 \vee x_3))\ \ ) \vee ((\neg x_1) \wedge x_4)$$

Question 7.2 Use the backtracking algorithm to find whether or not the following formula is satisfiable. If it is satisfiable, give a satisfying assignment.

$$((\neg x_1) \vee x_2) \wedge (x_2 \vee x_3) \wedge (x_3 \vee x_4) \wedge (x_4 \vee (\neg x_1)) \wedge (x_1 \vee (\neg x_3)) \wedge ((\neg x_2) \vee (\neg x_4))$$

Question 8 For each subquestion, check $\boxed{\checkmark}$ the correct answer. No justification necessary.

Consider a neural network with $\Gamma > 0$ neurons, arranged in $\ell > 0$ layers. The output of the neural network is given by a function $F : \mathbb{R}^n \to \mathbb{R}^k$ (i.e. $F(x) \in \mathbb{R}^k$ for some $x \in \mathbb{R}^n$). We train the neural network with a training set of vectors x^j and associated categories c^j, for $j = 1, \ldots, J$.

1. The neural network classifies its inputs into categories. How many categories?

☐ (a) n categories.
☐ (b) k categories.
☐ (c) Γ/ℓ categories.
☐ (d) J categories.

2. If we are training the neural network by minimizing the cost function

$$\frac{1}{J}\sum_{j=1}^{J}\frac{1}{2}\left|\left|e_{c^j} - F(x^j)\right|\right|^2$$

where F is computed using the following relation for each neuron: the output $y^i \in \mathbb{R}$ of a neuron $i \in \{1, \ldots, \Gamma\}$ with input $z^i \in \mathbb{R}^g$ is

$$y^i = \sigma\left((w^i)^T z^i + b^i\right).$$

What are the variables in the optimization problem?

☐ (a) x^j for all $j \in \{1, \ldots, J\}$.
☐ (b) σ.
☐ (c) w^i and b^i for all $i \in \{1, \ldots, \Gamma\}$.
☐ (d) y^i and z^i for all $i \in \{1, \ldots, \Gamma\}$.

3. In *stochastic* gradient descent, the gradient of the cost function is approximated is approximated by computing

☐ (a) $e_s^T \frac{1}{J}\sum_{j=1}^{J}\frac{1}{2}\nabla\left|\left|e_{c^j} - F(x^j)\right|\right|^2$ for $s \in S$, $S \subseteq \{1, \ldots, p\}$ where p is the number of variables.
☐ (b) $\frac{1}{J}\sum_{j=1}^{J}\frac{1}{2}\nabla\sum_{s\in S}\left|e_s^T(e_{c^j} - F(x^j))\right|^2$ where $S \subseteq \{1, \ldots, k\}$.
☐ (c) $\frac{1}{J}\sum_{j=1}^{J}\frac{1}{2}\nabla\sum_{s\in S}\left|e_{c^j} - F(e_s^T x^j))\right|^2$ where $S \subseteq \{1, \ldots, n\}$.
☐ (d) $\frac{1}{|S|}\sum_{j\in S}\frac{1}{2}\nabla\left|\left|e_{c^j} - F(x^j)\right|\right|^2$ where $S \subseteq \{1, \ldots, J\}$.

4. Given an input $\bar{x}$ and the corresponding output $F(\bar{x})$ of the neural network, with values

$$\bar{x} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \\ \bar{x}_4 \\ \bar{x}_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \qquad F(\bar{x}) = \begin{bmatrix} F_1(\bar{x}) \\ F_2(\bar{x}) \\ F_3(\bar{x}) \\ F_4(\bar{x}) \\ F_5(\bar{x}) \\ F_6(\bar{x}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0.1 \\ 0.8 \\ -0.2 \\ 0 \end{bmatrix},$$

what is the predicted category for $\bar{x}$?

☐ (a) Category 1.

☐ (b) Category 2.

☐ (c) Category 3.

☐ (d) Category 4.

☐ (e) Category 5.