SIDRA EXAM 2023

SURNAME and NAME: ____

Solve exactly three exercises: at least one about Optimization-Module 1 (Prof. Monaci) and at least one about Optimization-Module 2 (Prof. Cacchiani) by choosing from the list below.

You can do the three exercises in a file or on paper. In any case, you have to deliver a pdf file (you can scan your paper) via email to Prof. Monaci (michele.monaci@unibo.it) and Prof. Cacchiani (valentina.cacchiani@unibo.it).
Remember to write your name and surname in the pdf file.

MODULE 1 - Prof. Monaci

Exercise 1.1

Let $f : \Re^2 \to \Re, x \to 2x_1^4 x_2 + x_2^2 x_1$ be a function to be minimized.
Apply the gradient method to determine a descendant direction (if any) for each of the following points: A = (0, 1), B = (0, 0), and C = (1, 0). For point A, determine the best step size and the next point that would be computed by the algorithm.

Exercise 1.2

Consider the following constrained optimization problem

$$\min x_1^2 + 3x_2 \quad h(x) = x_1^2 - x_2^2 - 1 = 0$$

Define the Lagrangian relaxation of the problem and use it to compute a local optimum.

MODULE 2 - Prof. Cacchiani

Exercise 2.1

Solve the following Knapsack Problem by applying the Dynamic Programming Algorithm. Report the table, as we did in the lectures, with the maximum profit values computed for each subset of items and for each value of capacity.
Report the final optimal solution value and the set of items that were selected to be inserted in the knapsack in the optimal solution.

Knapsack Problem:

- number of items $n = 5$;
- capacity of the knapsack $C = 15$;
- weight vector $w = (7, 5, 6, 3, 5)$;
- profit vector $p = (5, 5, 7, 4, 6)$.

Exercise 2.2

Consider the following Integer Linear Programming model and solve it by applying the Branch and Bound Algorithm. At each node of the branch decision tree, compute the bound by solving the continuous relaxation of the corresponding subproblem through the graphical method. For branching select the fractional variable with the **smallest index** and first explore nodes derived by $\geq$ **condition** in **depth first** strategy.

For each explored node, draw the feasible region corresponding to the associated continuous relaxation (one drawing for each node). Draw the branch decision tree and report, next to each node, the solution of the continuous relaxation, the corresponding value and the corresponding bound. Report the optimal solution found and its value.

$$\begin{gathered}
\min \ 2x_1 + x_2 \\
2x_1 \geq 1 \\
2x_2 \geq 1 \\
2x_1 + 2x_2 \geq 3 \\
x_1, x_2 \geq 0, \quad \textit{integer}
\end{gathered}$$

Exercise 2.3

Given the Knapsack Problem formulated as in the following model, write a Light Robust model for it by considering uncertainty in the item weights. You can add the parameters needed in the Light Robust model.

Notation:
N indicates the set of items and C is the knapsack capacity.
p_j is the profit of item $j \in N$ and w_j is the weight of item $j \in N$.
The binary variable x_j is 1 if item j is selected for being inserted in the knapsack and 0 otherwise ($j \in N$).

$$\begin{aligned} &\max \textstyle\sum_{j \in N} p_j x_j \\ &\textstyle\sum_{j \in N} w_j x_j \leq C \\ &x_j \in \{0,1\} \quad j \in N \end{aligned}$$

Exercise 2.4

Consider the following problem called the *Generalized Assignment Problem* (GAP).

In this problem, there are n tasks and m agents. Each agent i has capacity b_i, each task j uses capacity a_{ij} of agent i. For example, a_{ij} is the time required by agent i to execute task j and b_i is the total working time of agent i ($i = 1, \dots, m$, $j = 1, \dots, n$).

The cost of assigning task j to agent i is c_{ij} ($i = 1, \dots, m$, $j = 1, \dots, n$).

The problem requires to assign each task to exactly one agent at minimum cost subject to capacity constraints.

The binary variable x_{ij} is 1 if task j is assigned to agent i and 0 otherwise ($i = 1, \dots, m$, $j = 1, \dots, n$). The following model represents a valid formulation for the GAP.

$$\begin{aligned} &\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\ &\sum_{j=1}^{n} a_{ij} x_{ij} \leq b_i && i = 1, \dots, m \\ &\sum_{i=1}^{m} x_{ij} = 1 && j = 1, \dots, n \\ &x_{ij} \in \{0,1\} && i = 1, \dots, m \; j = 1, \dots, n \end{aligned}$$

Improve the formulation above by introducing cover inequalities to replace the capacity constraints.