

Linear Optimisation Assignment-2: Implementing the Simplex Method

Due: 17 March

1 Objective

The goal of this assignment is to implement the Simplex method for solving linear programming problems in standard form. You are required to code the Simplex algorithm from scratch without using any inbuilt or external solvers.

2 Assignment Details

- You must implement a Python function with the following signature:

```
def Simplex(A, b, c):  
    Optimal = None  
    X = None  
    return (X, Optimal)
```

This function should solve the linear program in standard form:

$$\begin{array}{ll}\text{minimize} & c^T x \\ \text{subject to:} & Ax = b \\ & x \geq 0\end{array}$$

- You are permitted to use only basic Python libraries such as `numpy`, `os` and `pickle`. Using any inbuilt solvers (for example, those available in Gurobi, SciPy) is strictly prohibited.
- The value of `n` may be around 5 or 10. This should not be an issue; try to write a generalized code that can take any valid `A`, `b`, and `c`.
- The provided boilerplate code in `code_RollNumber.py` loads the test data from a pickle file (`input_data.pkl`), calls your `Simplex` function, and saves the result in a new pickle file. The output filename is created by appending “`_solution`” to your code file’s name.

3 Submission Guidelines

1. **Python Code File:** Submit your Python code file (e.g., `code_RollNumber.py`) containing your implementation of the Simplex method. Replace the word “RollNumber” with your actual roll number. This will map your file to your profile during automated testing.
2. **Naming Convention & File Structure: DO NOT change the naming convention.** Please follow the file structure exactly as mentioned below:
 - `A2_RollNumber.zip`
 - `code_RollNumber.py`
 - Other files if needed (although, typically, no additional files are necessary!)
3. You have been provided with a `tester.py` file, kindly make your solution compatible with the testing script before submitting.
4. Presentations for the assignment would be held. During the presentation, you must be able to explain your code and the rationale behind your approach. Your presentation will be used to evaluate the originality of your work. **Suspected cases of copying will be flagged and sent to Sir for further action!**

4 Important Notes

- **Presentations:** Presentations would be conducted for the Assignments, during which you would be asked to explain parts of your code.
- **Presentation:** Your ability to clearly explain your code during the presentation is a critical component of your evaluation.
- **Proper Naming and Error Handling:** If a code file does not run due to poor naming conventions or incorrect error handling, then the submission shall be voided.
- **Simplex Approach:** There are various versions, tweaking or approaches of implementing Simplex, you may implement any valid one.
- **Test Cases:** Some Test Cases, would be for infeasible problems. For such cases output (None, Infeasible), This is mentioned in the boiler plate too.

5 Evaluation Criteria

Your assignment will be evaluated based on:

- Correctness of your Simplex implementation, by the number of Test Cases it passes.

- Adherence to the assignment guidelines.
- Ability to explain your approach.
- Originality of your Submission.