ASSIGNMENT-2

Annam Deddeepya Purnima (cs22btech11006)
Mane Pooja Vinod (cs22btech11035)
Surbhi (cs22btech11057)

Question:

Assumption

1. Polytope is non-degenerate.

2. Rak of A is n

Implement the simplex algorithm to maximize the objective function, You need to implement the method discussed in class.

Input: CSV file with m+2 rows and n+1 column.

- **The first row excluding the last element is the initial feasible point z of length n**
- **The second row excluding the last element is the cost vector c of length n**
- **The last column excluding the top two elements is the constraint vector b of length m**
- **Rows third to m+2 and column one to n is the matrix A of size m*n**

Output: You need to print the sequence of vertices visited and the value of the objective function at that vertex

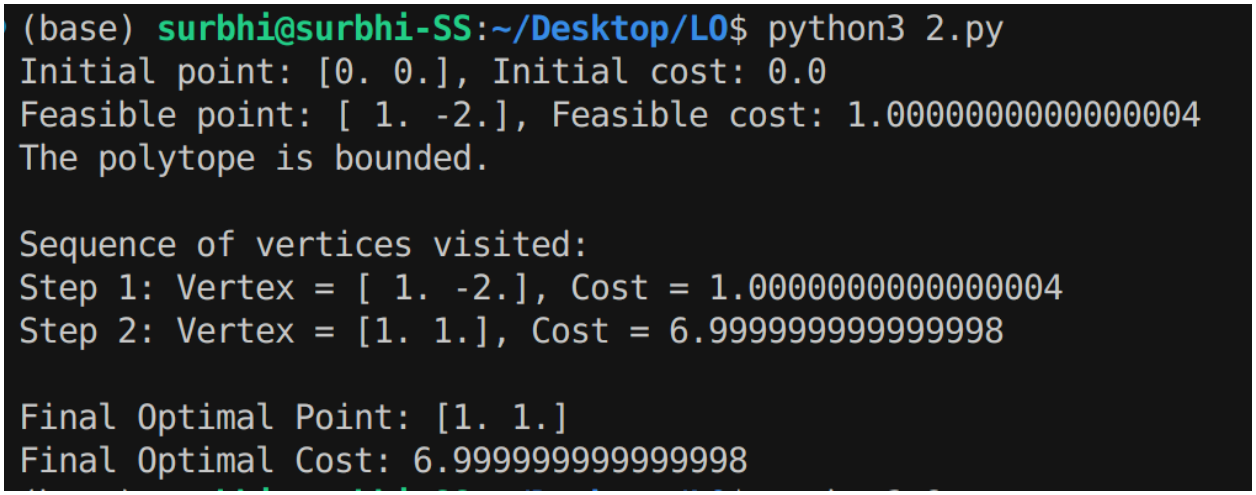
REPORT

Code implements the **Simplex algorithm** for solving linear programming problems. It takes a set of linear constraints and a cost function to minimize, adjusts an initial point to a feasible solution if necessary, and then optimizes the cost iteratively while checking feasibility and boundedness.

- ➔ Parsing the input file to extract:
  - Initial point **(z)**
  - Cost vector **(c)**
  - Constraints matrix **(A)** and vector **(v)**
- ➔ The **is_feasible_point** function checks if the given (z) satisfies the constraint $A \cdot z \leq v$

- The **adjust_to_basic_feasible** function adjusts (z) to a basic feasible point by iterating over active constraints using null-space computations.Handles scenarios where (z) is not already feasible.
- The **simplex** function iteratively moves between feasible vertices, reducing and optimizing the cost function **(c.z)** , using active constraints and reduced costs. Iterates until no negative reduced costs are found, indicating an optimal solution
- **check_bounded(A,v)** function determines if the polytope defined by $A \cdot z \leq v$ is bounded using **scipy.optimize.linprog.**
- We took **input as t.csv at line 125**, we can change according to our test cases.

OUTPUT:



```
(base) surbhi@surbhi-SS:~/Desktop/LO$ python3 2.py
Initial point: [0. 0.], Initial cost: 0.0
Feasible point: [ 1. -2.], Feasible cost: 1.0000000000000004
The polytope is bounded.

Sequence of vertices visited:
Step 1: Vertex = [ 1. -2.], Cost = 1.0000000000000004
Step 2: Vertex = [1. 1.], Cost = 6.999999999999998

Final Optimal Point: [1. 1.]
Final Optimal Cost: 6.999999999999998
```