



Università
della
Svizzera
italiana

Institute of
Computing
CI

Numerical Computing

2023

Student: Harkeerat Singh Sawhney

Solution for Project 5

Due date: Friday, December 29, 2023, 11:59 PM

Numerical Computing 2023 — Submission Instructions

(Please, notice that following instructions are mandatory:
submissions that don't comply with, won't be considered)

- Assignments must be submitted to iCorsi (i.e. in electronic format).
- Provide both executable package and sources (e.g. C/C++ files, MATLAB). If you are using libraries, please add them in the file. Sources must be organized in directories called:
Project_number_lastname_firstname
and the file must be called:
project_number_lastname_firstname.zip
project_number_lastname_firstname.pdf
- The TAs will grade your project by reviewing your project write-up, and looking at the implementation you attempted, and benchmarking your code's performance.
- You are allowed to discuss all questions with anyone you like; however: (i) your submission must list anyone you discussed problems with and (ii) you must write up your submission independently.

1. Graphical Solution of Linear Programming Problems [20 points]

1. Problem 1

1.1. Solve the System of Inequalities

We are asked to solve the system of inequalities, which is already given in the question itself.

$$\begin{aligned} \min \quad & z = 4x + y \\ \text{s.t.} \quad & x + 2y \leq 40 \\ & x + y \geq 30 \\ & 2x + 3y \geq 72 \\ & x, y \geq 0 \end{aligned} \tag{1}$$

Hence as it can be seen from the Equation 1, we have to find the minimum value of z such that the constraints are satisfied. This can be solved graphically by plotting the constraints and finding the minimum value of z . Before that lets rewrite the constraints in the form of $y = mx + c$ which can be seen at Equation 2.

$$\begin{aligned} \min \quad & z = 4x + y \\ \text{s.t.} \quad & y = -\frac{1}{2}x + 20 \\ & y = -x + 30 \\ & y = -\frac{2}{3}x + 24 \\ & x, y \geq 0 \end{aligned} \tag{2}$$

1.2. Plot the feasible region identified by the constraints.

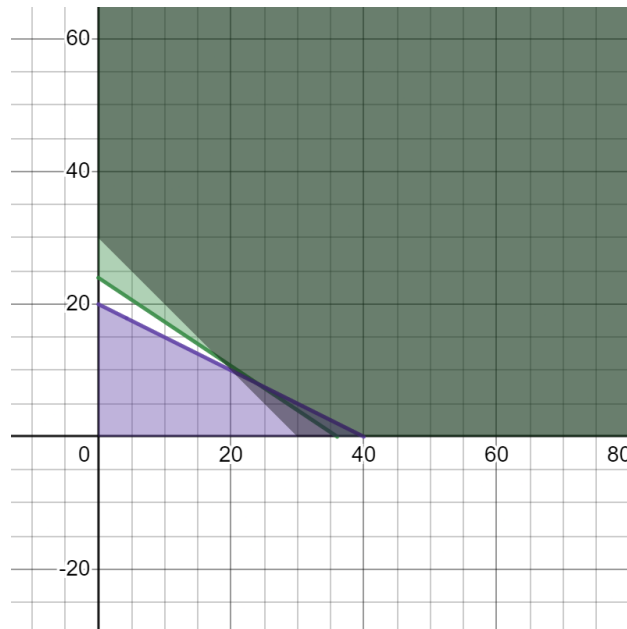


Figure 1: Plot of all the constraints for Problem 1

Now we can plot the constraints on a graph as shown in Figure 1. Figure 1 is showing the plot of all the constraints. All of these constraints can be seen by their respective colors. However what we are interested is in the region where all the constraints maximum, as that region would include all minimum value of z .

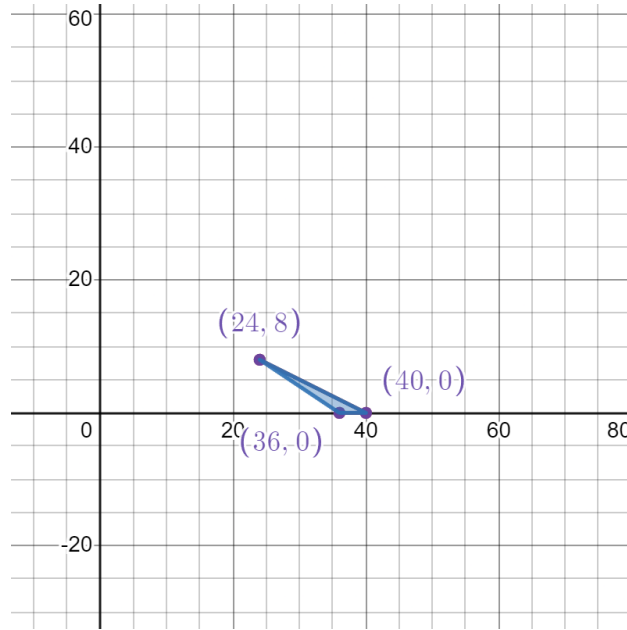


Figure 2: Shading the region where all the constraints maximum for Problem 1

As it can be seen from the Figure 2, the region where all the constraints maximum is the region where the minimum value of z would be. Here what is important for us is the points of intersection of the constraints with each other and the points of intersection of the constraints with the axes. These points are as well shown in the Figure 2. These points are as follows:

- (24, 8)
- (40, 0)
- (36, 0)

1.3. Find the optimal solution and the value of the objective function in that point.

Now from the previous part we have the points of intersection of the constraints with each other and the points of intersection of the constraints with the axes. What we need to do now is to find the minimum value of z at these points. Hence we have to evaluate the objective function at each vertices and pick the minimum value of z . The vertices are as follows:

$$\begin{aligned}
 \text{At } (24, 8) \quad z &= 4(24) + 8 = 104 \\
 \text{At } (40, 0) \quad z &= 4(40) + 0 = 160 \\
 \text{At } (36, 0) \quad z &= 4(36) + 0 = 144
 \end{aligned} \tag{3}$$

From the Equation 3 we can see that the minimum value of z is at (24, 8) which is 104.

2. Problem 2

2.1. Solve the System of Inequalities

This is a bit more tricky than the previous problem, as unlike previous problem we do now have the mathematical form of the constraints. We have a real life problem from which we have to convert to the mathematical form. The problem is as follows:

A tailor plans to sell two types of trousers, with production costs of 25 CHF and 40 CHF, respectively. The former type can be sold for 85 CHF, while the latter for 110 CHF. The tailor estimates a total monthly demand of 265 trousers. Find the number

of units of each type of trousers that should be produced in order to maximize the net profit of the tailor, if we assume that he cannot spend more than 7000 CHF in raw materials.?

Hence from the above problem we can see that we have to find the maximum value of z which is the net profit of the tailor. The variables are as follows:

- x = Number of trousers of type 1
- y = Number of trousers of type 2
- z = Net profit of the tailor

We have obtained the Objective Function (Maximize Profit z) by knowing that the profit from selling the first type of trousers is $85 - 25 = 60$ CHF and the profit from selling the second type of trousers is $110 - 40 = 70$ CHF. Hence we can write the Objective Function as follows:

$$\text{Maximize } z = 60x + 70y \quad (4)$$

We know that the total demand of trousers is 265 and the total amount of money that can be spent on raw materials is 7000 CHF. Hence we can write the first constraint as follows:

$$x + y \leq 265 \quad (5)$$

We also know that the cost of producing the first type of trousers is 25 CHF and the cost of producing the second type of trousers is 40 CHF. Hence we can write the second constraint as follows:

$$25x + 40y \leq 7000 \quad (6)$$

Hence finally by combining the Equation 4, 5 and 6 we can write the constraints as follows in the mathematical form at Equation 7.

$$\begin{aligned} \min \quad & z = 60x + 25y \\ \text{s.t.} \quad & x + y \leq 265 \\ & 25x + 40y \leq 7000 \\ & x, y \geq 0 \end{aligned} \quad (7)$$

Now we must rewrite the constraints in the form of $y = mx + c$ which can be seen at Equation 8.

$$\begin{aligned} \min \quad & z = 60x + 25y \\ \text{s.t.} \quad & y = -x + 265 \\ & y = -\frac{5}{8}x + 175 \\ & x, y \geq 0 \end{aligned} \quad (8)$$

2.2. Plot the feasible region identified by the constraints.

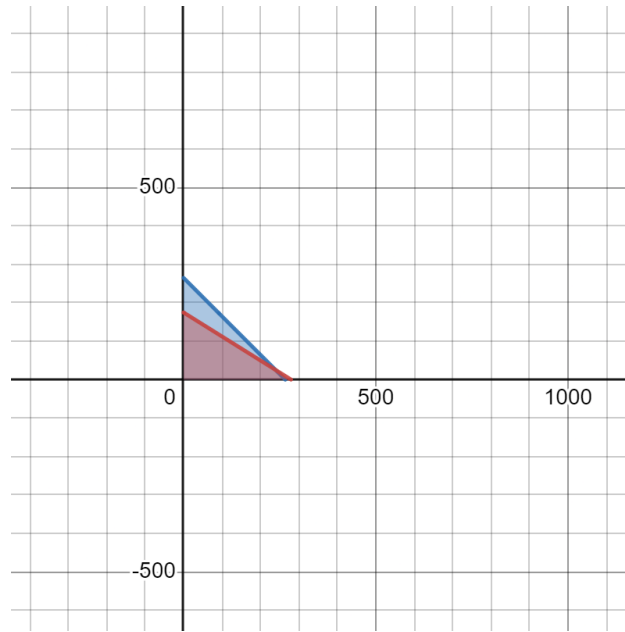


Figure 3: Plot of all the constraints for Problem 2

Again very similar to the previous problem, we have to plot the constraints on a graph as shown in Figure 3. Figure 3 is showing the plot of all the constraints and all of these constraints can be seen by their respective colors. However what we are interested is in the region where all the constraints maximum, as that region would include all maximize value of z .

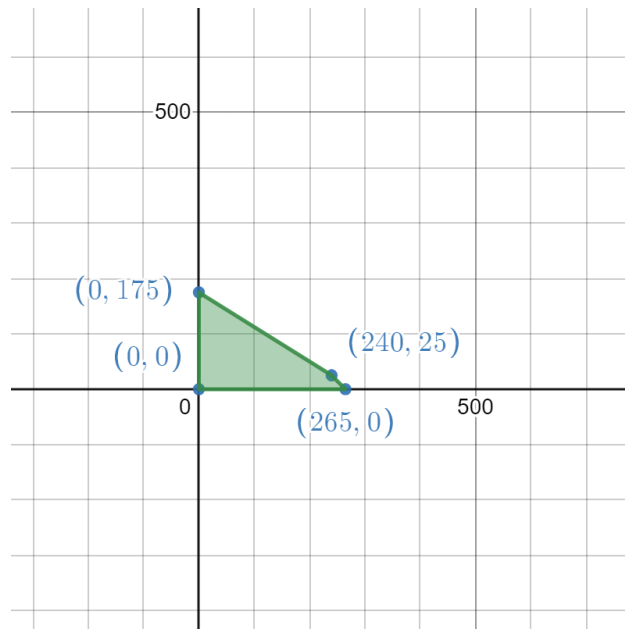


Figure 4: Shading the region where all the constraints maximum for Problem 2

As it can be seen from the Figure 4, the region where all the constraints maximum is the region where the maximum value of z would be. Here what is important for us is the points of intersection of the constraints with each other and the points of intersection of the constraints with the axes. These points are as well shown in the Figure 4. These points are as follows:

- $(0, 175)$

- (240, 25)
- (265, 0)
- (0, 0)

2.3. Find the optimal solution and the value of the objective function in that point.

Now from the previous part we have the points of intersection of the constraints with each other and the points of intersection of the constraints with the axes. What we need to do now is to find the minimum value of z at these points. Hence we have to evaluate the objective function at each vertex and pick the minimum value of z . The vertices are as follows:

$$\begin{aligned}
 \text{At } (0, 175) \quad z &= 60(0) + 25(175) = 4375 \\
 \text{At } (240, 25) \quad z &= 60(240) + 25(25) = 15025 \\
 \text{At } (265, 0) \quad z &= 60(265) + 25(0) = 15900 \\
 \text{At } (0, 0) \quad z &= 60(0) + 25(0) = 0
 \end{aligned} \tag{9}$$

From the Equation 9 we can see that the maximum value of z is at (265, 0) which is 15900.

2. Implementation of the Simplex Method [30 points]

In this question we are asked to implement the Simplex Method for the Linear Programming Problem. We are asked to complete the 2 functions which are as follows:

- `standardise.m`
- `simplexSolve.m`

Before explaining the implementation of the functions, I will first explain how the Simplex Method works. The Simplex Method is an iterative method for solving the Linear Programming Problems, in which it first starts with a feasible solution and then iteratively improves the solution until the optimal solution is reached. This is what the Matlab implementation of the Simplex Method is doing as well.

1. `standardise.m`

This function is used to standardize the Linear Programming Problem which is used to a form which can be used by other methods. This function takes in total 6 arguments which are as follows:

- `type` = A string which tells whether the problem is a maximization or minimization problem
- `A` = A matrix of the coefficients of the constraints
- `h` = The right hand side of the constraints
- `c` = The coefficients of the objective function
- `m` = The number of constraints
- `sign` = A vector indicating the sign of the constraints

1.1. Summery of the function

The function first initializes an identity matrix of size $m \times m$. Then it checks if the problem is a maximization or minimization problem. If the problem is a maximization problem then it uses the surplus variables (changes the sign of the corresponding row in the identity matrix), however if it is a minimization problem then it uses the slack variables (does not change the sign of the corresponding row in the identity matrix).

The next step is to ensure that all the values in the right-hand side (h) are positive. If any of the values are negative then it changes the sign of the corresponding row in A, h and the augmented matrix. This is done because the Simplex Method only works for the problems where the right-hand side is positive. This was the To-Do which we were supposed to implement, which was done by creating a for-loop which checks if any of the values in h is less than 0 and if it is, it multiplies the corresponding row in A, h and the augmented matrix by -1 . After that the functions arguments are returned.

2. simplexSolve.m

This function takes the following arguments:

- **type** = A string which tells whether the problem is a maximization or minimization problem
- **A** = A matrix of the coefficients of the constraints
- **h** = The right hand side of the constraints
- **c** = The coefficients of the objective function
- **sign** = A vector indicating the sign of the constraints

What this functions is supposed to do is to standardize the problem using the standardize function (which was explained in the previous section) and then determine a optimal starting solution which uses auxiliary function. Finally it solves the linear programming problem using the simplexSolve function and computes the value of the objective function at the optimal solution.

2.1. Summery of the function

First the function calculates the number of constraints m and the number of variables n from the dimensions of A . It also calculates the *itMax* which is the maximum number of iteration which the Simplex Method can do. This was done by using taking in the number of constraints and the number of variables and then multiplying them by 2. This was done because the maximum number of iterations which the Simplex Method can do is $(m+n)chosem$. In the actual code (which we were supposed to implement) we first calculated the factorial of $(m+n)$ and then divided it by the factorial of m and factorial of n . With this we calculated the binomial coefficient of $(m+n)chosem$.

After this it computes the standard form of the problem using the standardize function. Then it determines a starting solution using the auxiliary function. Finally it solves the linear programming problem using the simplexSolve function and computes the value of the objective function at the optimal solution. After that it computes the value of the function by multiplying the coefficient of the objective function with the optimal solution. Finally it returns the optimal solution and the value of the objective function.

3. simplexSolve.m

This function has the implementation of the Simplex Method. It first initializes the number of iteration $nIter$ to 0. It as well computes the $BB^{-1} * D$ and $B^{-1} * h$, which are the initial values of the Simplex Method. After this it computes the reduce coefficients (which we were supposed to implement). This was done by using the formula $r_d = c_d - c_B * (B \setminus D)$ These coefficients are used to determine which of these variables should be added to the basis. Then this function checks

if it is an optimal solution or not. If it is a maximization problem, then the solution would be optimal if all the reduced cost coefficients are non-positive. However, if it is a minimization problem, the solution is optimal if all reduced cost coefficients are non-positive.

Now the main loop is formulated which keeps on going until the optimal condition is satisfied. In the loop the following is done in each iteration:

- Entering the Variable: If it is a maximization problem the variable with the largest positive reduced cost coefficient enters the basis. If it is a minimization problem., the variable with the smallest negative reduced cost coefficient enters the basis.
- Evaluation of the coefficient ratios: The variable is determined which would leave the basis. This is done by computing the coefficient ratios $B^{-1} * h$ to B^{-1} in for each basic variable and then it choses the variable with the smallest positive ratio.
- Update the Matrices: The matrices B , D , cost vectors and the indices are updated by swapping the entering and leaving variables.
- Recomputing the reduced cost coefficients: The reduced cost coefficients are recomputed using the formula $r_d = c_d - c_B * (B \setminus D)$.
- Checking the optimality condition: The optimality condition is checked again and if it is satisfied then the loop is terminated.
- Checking the Loop bound: If the number of iterations exceeds the maximum number of iterations then the loop is terminated.
- Computing the optimal solution: The optimal solution is computed using the formula $x_B = B \setminus h$.

Finally the optimal solution and the value of the objective function is returned.

At last we ran the `test_simplexSolve.m` it has shown that all the tests have passed. Hence we can say that the implementation of the Simplex Method is correct.

3. Applications to Real-Life Example: Cargo Aircraft [25 points]

1. **Formulate the problem as a linear program: what is the objective function? What are the constraints? Write down all equations, with comments explaining what you are doing.**

In this Question we are asked to solve a real life Linear Programming Problem. The problem is that we have a cargo aircraft which has 4 components and we have to find the optimal way to load the cargo in the aircraft. Bellow is the table which shows the compartments and their capacities.

Compartment	Weight Capacity (t)	Storage Capacity (m3)
S1	18	11930
S2	32	22552
S3	25	11209
S4	17	5870

Table 1: Compartments and their capacities

Cargo	Weight (t)	Volume (m3/t)	Profit (CHF/t)
C1	16	320	135
C2	32	510	200
C3	40	630	410
C4	28	125	520

Table 2: Cargos and their properties

As it can be seen in the Table 1 and 2 that we have 4 compartments and 4 cargos. The compartments have their weight capacity and storage capacity. The cargos have their weight, volume and profit. Along with that we also know that the profit obtained for each cargo increases by some percentage depending on the compartment in which it is stored. This percentage is as follows:

- $S1 = 0\%$
- $S2 = 10\%$
- $S3 = 20\%$
- $S4 = 30\%$

1.1. Objective Function

The objective function is to maximize the profit. The profit is obtained by multiplying the profit of each cargo with the percentage increase in the profit depending on the compartment in which it is stored. hence the objective function is as follows:

Lets first define the variables which we are going to use in the objective function. x_{ij} is the number of cargos of type i stored in compartment j . So for example x_{11} is the number of cargos of type 1 stored in compartment 1.

Objective Function

$$\text{Maximize } z = 135 \times 1 \times \sum_{j=1}^4 x_{1j} + 200 \times 1.1 \times \sum_{j=1}^4 x_{2j} + 410 \times 1.2 \times \sum_{j=1}^4 x_{3j} + 520 \times 1.3 \times \sum_{j=1}^4 x_{4j} \quad (10)$$

1.2. Constraints

The objective function is to maximize the profit. However we have some constraints which we have to follow in order to maximize the profit. These constraints are as follows:

- The weight of the cargo in each compartment should not exceed the weight capacity of the compartment.
- The volume of the cargo in each compartment should not exceed the volume capacity of the compartment.
- The total number of cargos of each type should not exceed the total number of cargos available.
- We need to follow the non-negativity constraint.

Bellow is the visualization of the constraints:

Weight Constraint

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &\leq 18 \\ x_{21} + x_{22} + x_{23} + x_{24} &\leq 32 \\ x_{31} + x_{32} + x_{33} + x_{34} &\leq 25 \\ x_{41} + x_{42} + x_{43} + x_{44} &\leq 17 \end{aligned} \quad (11)$$

Volume Constraint

$$\begin{aligned} 320x_{11} + 510x_{12} + 630x_{13} + 125x_{14} &\leq 11930 \\ 320x_{21} + 510x_{22} + 630x_{23} + 125x_{24} &\leq 22552 \\ 320x_{31} + 510x_{32} + 630x_{33} + 125x_{34} &\leq 11209 \\ 320x_{41} + 510x_{42} + 630x_{43} + 125x_{44} &\leq 5870 \end{aligned} \tag{12}$$

Number of Cargos Constraint

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &\leq 16 \\ x_{21} + x_{22} + x_{23} + x_{24} &\leq 32 \\ x_{31} + x_{32} + x_{33} + x_{34} &\leq 40 \\ x_{41} + x_{42} + x_{43} + x_{44} &\leq 28 \end{aligned} \tag{13}$$

Non-Negativity Constraint

$$x_{ij} \geq 0 \quad \forall i, j \tag{14}$$

2. Create a script `exercise2.m` which uses the simplex method implemented in the previous exercise to solve the problem. What is the optimal solution? Visualize it graphically and briefly comment the results obtained (are you surprised of this outcome on the basis of your data?).

By using the `simplexSolve.m` function which we implemented in the previous exercise, we can solve the problem. The code for the `exercise2.m` is as follows:

```
type = 'max';

A = [1,0,0,0, 1,0,0,0, 1,0,0,0, 1,0,0,0;
     0,1,0,0, 0,1,0,0, 0,1,0,0, 0,1,0,0;
     0,0,1,0, 0,0,1,0, 0,0,1,0, 0,0,1,0;
     0,0,0,1, 0,0,0,1, 0,0,0,1, 0,0,0,1;
     320,0,0,0, 510,0,0,0, 630,0,0,0, 125,0,0,0;
     0,320,0,0, 0,510,0,0, 0,630,0,0, 0,125,0,0;
     0,0,320,0, 0,0,510,0, 0,0,630,0, 0,0,125,0;
     0,0,0,320, 0,0,0,510, 0,0,0,630, 0,0,0,125;
     1,1,1,1, 0,0,0,0, 0,0,0,0, 0,0,0,0;
     0,0,0,0, 1,1,1,1, 0,0,0,0, 0,0,0,0;
     0,0,0,0, 0,0,0,0, 1,1,1,1, 0,0,0,0;
     0,0,0,0, 0,0,0,0, 0,0,0,0, 1,1,1,1];

h = [18; 32; 25; 17; 11930; 22552; 11209; 5870; 16; 32; 40; 28];

c = [135, 135*1.1, 135*1.2, 135*1.3, 200, 200*1.1, 200*1.2, 200*1.3, 410,
     410*1.1, 410*1.2, 410*1.3, 520, 520*1.1, 520*1.2, 520*1.3];

sign = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0];

[z, x_B, index_B] = simplex(type, A, h, c, sign);
fprintf('z := %d\n', z);
fprintf('x_B := \n');
disp(x_B);
fprintf('index_B := \n');
disp(index_B);

bar(x_B);
title('Solution - (x_B)');
xlabel('Index');
ylabel('Value');
```

As it can be seen from the code 2, we have used the `simplexSolve.m` function to solve the problem. However the trick part for this question was to formulate the problem in the form which can be used by the `simplexSolve.m` function. We first created the variable A which is the matrix of the coefficients of the constraints. Then we created the variable h which is the right hand side of the constraints. Then we created the variable c which is the coefficients of the objective function. Finally we created the variable $sign$ which is a vector indicating the sign of the constraints.

After running the script the maximum profit which we can obtain is $z = 41890$. From the data I did not believe that the profit will be as high as that, as the profit for each cargo is not that high. However the profit is high because of the percentage increase in the profit depending on the compartment in which it is stored. Since we got the optimal solution, we maximized our profit from those percentage increase.

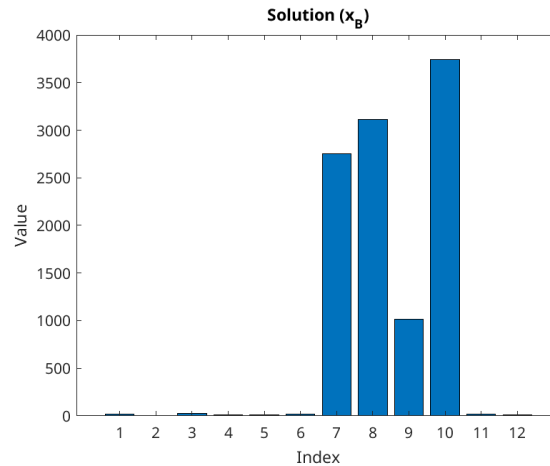


Figure 5: Plot of the x_B

The Figure 5 shows the plot of the x_B . x_B is the vector which contains the basic variables.

4. Cycling and Degeneracy [10 points]

I tried to implement this problem in the similar way as the previous question, however I am getting the error of saying `Incorrect loop, more iterations than the number of basic solutions`. I tried to debug the code, however I was not able to find the error. Hence I am submitting the code which I have implemented. The code is as follows:

```
type = 'max';
A = [4, 3; 4, 1; 4, 2];
h = [12; 8; 8];
c = [3, 4];
sign = [-1, -1, -1];

[z, x_B, index_B] = simplex(type, A, h, c, sign);
```