МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Дальневосточный федеральный университет»
(ДВФУ)

ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ НАУК

Кафедра информатики, математического и компьютерного моделирования

ОТЧЕТ

к работе №3 по дисциплине
«Методы оптимизации»

Направление подготовки
01.03.02 «Прикладная математика и информатика»

Выполнил студент
гр. Б9118-01.03.02миопд
Охроменко Д.А. ________
(ФИО) *(подпись)*

« 5 » января 2022 г.

г. Владивосток
2022

Постановка задачи

Линейная оптимизация. Оптимизация производства. Решение симплекс-методом задачи:

$$\begin{cases} c \cdot x \to max, \\ Ax \le b, \\ x \ge 0 \end{cases}$$

A - неотрицательная матрица 8×6,
x - неотрицательный шестимерный вектор,
b - неотрицательный восьмимерный вектор,
c - неотрицательный шестимерный вектор

$$A = \begin{pmatrix} 3 & 5 & 4 & 6 & 2 & 1 \\ 6 & 2 & 8 & 2 & 9 & 3 \\ 3 & 6 & 5 & 7 & 4 & 6 \\ 5 & 3 & 2 & 3 & 5 & 3 \\ 4 & 8 & 4 & 8 & 4 & 5 \\ 2 & 4 & 7 & 2 & 1 & 2 \\ 5 & 2 & 4 & 5 & 7 & 1 \\ 7 & 9 & 6 & 1 & 3 & 2 \end{pmatrix} \qquad b = \begin{pmatrix} 3 & 6 & 5 & 2 & 7 & 2 & 1 & 4 \end{pmatrix}$$

$$c = \begin{pmatrix} 1 & 8 & 5 & 7 & 9 & 3 \end{pmatrix}$$

Ищется оптимальное решение (симплекс-метод). В двойственной задаче ищется допустимое и оптимальное решение.

Прямая задача

Для решения прямой задачи воспользуемся симплекс-методом.

В начале нам необходимо привести задачу к каноническому виду, для этого вводим дополнительный восьмимерный вектор z:

$$z = b - Ax$$

Припишем справа к матрице A единичную матрицу I, к вектору c дописываем восемь нулей, а вектор x дополняем вектором z:

$$\begin{cases}(c,\ 0)\cdot\begin{pmatrix}x\\z\end{pmatrix}\to max,\\(A\mid I)\begin{pmatrix}x\\z\end{pmatrix}=b\\x,z\geq 0\end{cases}$$

Составим симплекс таблицу. Нулевая строка - расширенный вектор c, где элементы записываются со знаком минус, т.к. каноническая задача на минимум, а данная задача на максимум. Следующие строки содержат расширенную матрицу A. Последний столбец - вектор b. Нулевой элемент этого вектора - значение целевой функции. На первой итерации она равна 0.

-1	-8	-5	-7	$\boxed{-9}$	-3	0	0	0	0	0	0	0	0	0
3	5	4	6	2	1	1	0	0	0	0	0	0	0	3
6	2	8	2	9	3	0	1	0	0	0	0	0	0	6
3	6	5	7	4	6	0	0	1	0	0	0	0	0	5
5	3	2	3	5	3	0	0	0	1	0	0	0	0	2
4	8	4	8	4	5	0	0	0	0	1	0	0	0	7
2	4	7	2	1	2	0	0	0	0	0	1	0	0	2
5	2	4	5	7	1	0	0	0	0	0	0	1	0	1
7	9	6	1	3	2	0	0	0	0	0	0	0	1	4

Также нам необходимо начальное угловое решение. Симплекс-метод стартует с этого начального углового решения и дает нам дальнейшие шаги для поиска оптимального решения. Начальное угловое решение:

$$x = 0,\ z = b$$

$$\begin{pmatrix}0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 6 & 5 & 2 & 7 & 2 & 1 & 4\end{pmatrix}$$

На следующем шаге нам необходимо выбрать базисный столбец. Для этого находим максимальное по модулю число в нулевой строке, этот элемент выделен в таблице выше. Столбец с ним будет разрешающим.

Необходимо поделить элементы последнего столбца на элементы разрешающего и найти минимальное отношение. Элемент, который будет минимальным - разрешающий элемент, а строка, которая содержит этот элемент - разрешающая строка.

Делим разрешающую строку на разрешающий элемент, который станет равным 1. Затем с помощью элементарных преобразований строк обнуляем остальные элементы разрешающего столбца. Получаем угловое решение x. Процесс будет повторяться до тех пор, пока в нулевой строке присутсвуют отрицательные элементы.

Первая итерация:

-1	-8	-5	-7	-9	-3	0	0	0	0	0	0	0	0	0
3	5	4	6	2	1	1	0	0	0	0	0	0	0	3
6	2	8	2	9	3	0	1	0	0	0	0	0	0	6
3	6	5	7	4	6	0	0	1	0	0	0	0	0	5
5	3	2	3	5	3	0	0	0	1	0	0	0	0	2
4	8	4	8	4	5	0	0	0	0	1	0	0	0	7
2	4	7	2	1	2	0	0	0	0	0	1	0	0	2
5	2	4	5	7	1	0	0	0	0	0	0	1	0	1
7	9	6	1	3	2	0	0	0	0	0	0	0	1	4

Разрешающий столбец 5, разрешающая строка 7, разрешающий элемент 7.

Вторая итерация:

5.43	-5.43	0.14	-0.57	0	-1.71	0	0	0	0	0	0	1.29	0	1.29
1.57	4.43	2.86	4.57	0	0.71	1	0	0	0	0	0	-0.29	0	2.71
-0.43	-0.57	2.86	-4.43	0	1.71	0	1	0	0	0	0	-1.29	0	4.71
0.14	4.86	2.71	4.14	0	5.43	0	0	1	0	0	0	-0.57	0	4.43
1.43	1.57	-0.86	-0.57	0	2.29	0	0	0	1	0	0	-0.71	0	1.29
1.14	6.86	1.71	5.14	0	4.43	0	0	0	0	1	0	-0.57	0	6.43
1.29	3.71	6.43	1.29	0	1.86	0	0	0	0	0	1	-0.14	0	1.86
0.71	0.29	0.57	0.71	1	0.14	0	0	0	0	0	0	0.14	0	0.14
4.86	8.14	4.29	-1.14	0	1.57	0	0	0	0	0	0	-0.43	1	3.57

Разрешающий столбец 2, разрешающая строка 8, разрешающий элемент 8.14.

Третья итерация:

8.67	0	3	-1.33	0	-0.67	0	0	0	0	0	0	1	0.67	3.67
-1.07	0	0.53	5.19	0	-0.14	1	0	0	0	0	0	-0.05	-0.54	0.77
-0.09	0	3.16	-4.51	0	1.82	0	1	0	0	0	0	-1.32	0.07	4.96
-2.75	0	0.16	4.82	0	4.49	0	0	1	0	0	0	-0.32	-0.6	2.3
0.49	0	-1.68	-0.35	0	1.98	0	0	0	1	0	0	-0.63	-0.19	0.6
-2.95	0	-1.89	6.11	0	3.11	0	0	0	0	1	0	-0.21	-0.84	3.42
-0.93	0	4.47	1.81	0	1.14	0	0	0	0	0	1	0.05	-0.46	0.23
0.54	0	0.42	0.75	1	0.09	0	0	0	0	0	0	0.16	-0.04	0.02
0.6	1	0.53	-0.14	0	0.19	0	0	0	0	0	0	-0.05	0.12	0.44

Разрешающий столбец 4, разрешающая строка 7, разрешающий элемент 0.75.

Четвертая итерация:

9.63	0	3.74	0	1.77	-0.51	0	0	0	0	0	0	1.28	0.6	3.7
-4.81	0	-2.37	0	-6.88	-0.74	1	0	0	0	0	0	-1.14	-0.38	0.65
3.16	0	5.67	0	5.98	2.35	0	1	0	0	0	0	-0.37	-0.14	5.07
-6.23	0	-2.53	0	-6.4	3.93	0	0	1	0	0	0	-1.33	-0.37	2.19
0.74	0	-1.49	0	0.465	2.02	0	0	0	1	0	0	-0.56	-0.21	0.6
-7.35	0	-5.3	0	-8.09	2.4	0	0	0	0	1	0	-1.49	-0.56	3.28
-2.23	0	3.47	0	-2.4	0.93	0	0	0	0	0	1	-0.33	-0.37	0.19
0.72	0	0.56	1	1.33	0.12	0	0	0	0	0	0	0.21	-0.05	0.02
0.7	1	0.6	0	0.19	0.21	0	0	0	0	0	0	-0.02	0.12	0.44

Разрешающий столбец 6, разрешающая строка 6, разрешающий элемент 0.93.

Пятая итерация:

8.4	0	5.65	0	0.45	0	0	0	0	0	0	0.55	1.1	0.4	3.8
-6.6	0	0.4	0	-8.8	0	1	0	0	0	0	0.8	-1.4	-0.6	0.8
8.8	0	-3.07	0	12.02	0	0	1	0	0	0	-2.5	0.45	0.8	4.6
3.2	0	-17.18	0	3.73	0	0	0	1	0	0	-4.23	0.05	1.2	1.4
5.6	0	-9.03	0	5.68	0	0	0	0	1	0	-2.18	0.15	0.6	0.2
-1.6	0	-14.23	0	-1.93	0	0	0	0	0	1	-2.58	-0.65	0.4	2.8
-2.4	0	3.72	0	-2.58	1	0	0	0	0	0	1.08	-0.35	-0.4	0.2
1	0	0.13	1	1.63	0	0	0	0	0	0	-0.15	0.25	0	0
1.2	1	-0.18	0	0.73	0	0	0	0	0	0	-0.23	0.05	0.2	0.4

Отрицательные элементы в первой строке отсутсвуют. Оптимальное решение:

$$\begin{pmatrix} 0 & 0.4 & 0 & 0 & 0 & 0.2 \end{pmatrix}$$

Целевая функция равна 3.8

Двойственная задача

Двойственная задача имеет вид:

$$\begin{cases} b \cdot y \to min, \\ A^T y \geq c \\ y \geq 0 \end{cases}$$

b - неотрицательный восьмимерный вектор,
y - неотрицательный восьмимерный вектор,
c - неотрицательный шестимерный вектор.

$$A^T = \begin{pmatrix} 3 & 6 & 3 & 5 & 4 & 2 & 5 & 7 \\ 5 & 2 & 6 & 3 & 8 & 4 & 2 & 9 \\ 4 & 8 & 5 & 2 & 4 & 7 & 4 & 6 \\ 6 & 2 & 7 & 3 & 8 & 2 & 5 & 1 \\ 2 & 9 & 4 & 5 & 4 & 1 & 7 & 3 \\ 1 & 3 & 6 & 3 & 5 & 2 & 1 & 2 \end{pmatrix} \qquad b = \begin{pmatrix} 3 & 6 & 5 & 2 & 7 & 2 & 1 & 4 \end{pmatrix}$$

$$c = \begin{pmatrix} 1 & 8 & 5 & 7 & 9 & 3 \end{pmatrix}$$

Для решения задачи также воспользуемся симплекс-методом. Приведем ее к каноническому виду, введя шестимерный вектор z:

$$z = A^T y - c.$$

Добавим к матрице A^T отрицательную единичную матрицу I, вектор b дополним нулевым вектором, а вектор y вектором z:

$$\begin{cases} (b,\ 0) \cdot \begin{pmatrix} y \\ z \end{pmatrix} \to min, \\ (A \mid -I) \begin{pmatrix} y \\ z \end{pmatrix} = c \\ y, z \geq 0 \end{cases}$$

В этой задаче у нас отсуствует начальное угловое решение. Для его нахождения необходимо решить вспомогательную задачу, введем восьмимерный вектор u :

$$\begin{cases} \sum_{i=1}^{m} u_i \to min, \\ (A^T| - I|\, I)\begin{pmatrix} y \\ z \\ u \end{pmatrix} = c, \\ y, z, u \geq 0 \end{cases}$$

Оптимальное решение вспомогательной задачи является начальным решением двойственной задачи.

Построим таблицу:

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0
3	6	3	5	4	2	5	7	-1	0	0	0	0	0	1	0	0	0	0	0	1
5	2	6	3	8	4	2	9	0	-1	0	0	0	0	0	1	0	0	0	0	8
4	8	5	2	4	7	4	6	0	0	-1	0	0	0	0	0	1	0	0	0	5
6	2	7	3	8	2	5	1	0	0	0	-1	0	0	0	0	0	1	0	0	7
2	9	4	5	4	1	7	3	0	0	0	0	-1	0	0	0	0	0	1	0	9
1	3	6	3	5	2	1	2	0	0	0	0	0	-1	0	0	0	0	0	1	3

Перепишем таблицу, используя элементарные преобразования:

-21	-30	-31	-21	-33	-18	-24	-28	1	1	1	1	1	1	0	0	0	0	0	0	-33
3	6	3	5	4	2	5	7	-1	0	0	0	0	0	1	0	0	0	0	0	1
5	2	6	3	8	4	2	9	0	-1	0	0	0	0	0	1	0	0	0	0	8
4	8	5	2	4	7	4	6	0	0	-1	0	0	0	0	0	1	0	0	0	5
6	2	7	3	8	2	5	1	0	0	0	-1	0	0	0	0	0	1	0	0	7
2	9	4	5	4	1	7	3	0	0	0	0	-1	0	0	0	0	0	1	0	9
1	3	6	3	5	2	1	2	0	0	0	0	0	-1	0	0	0	0	0	1	3

Начальное угловое решение задачи:

$$y = 0,\ z = 0,\ u = c$$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 8 & 5 & 7 & 9 & 3 \end{pmatrix}$$

Первая итерация:

-21	-30	-31	-21	-33	-18	-24	-28	1	1	1	1	1	1	0	0	0	0	0	0	-33
3	6	3	5	4	2	5	7	-1	0	0	0	0	0	1	0	0	0	0	0	1
5	2	6	3	8	4	2	9	0	-1	0	0	0	0	0	1	0	0	0	0	8
4	8	5	2	4	7	4	6	0	0	-1	0	0	0	0	0	1	0	0	0	5
6	2	7	3	8	2	5	1	0	0	0	-1	0	0	0	0	0	1	0	0	7
2	9	4	5	4	1	7	3	0	0	0	0	-1	0	0	0	0	0	1	0	9
1	3	6	3	5	2	1	2	0	0	0	0	0	-1	0	0	0	0	0	1	3

Разрешающий столбец 5, разрешающая строка 1, разрешающий элемент 4.

Вторая итерация:

3.75	19.5	-6.25	20.25	0	-1.5	17.25	29.75	-7.25	1	1	1	1	1	8.25	0	0	0	0	0	-24.75
0.75	1.5	0.75	1.25	1	0.5	1.25	1.75	-0.25	0	0	0	0	0	0.25	0	0	0	0	0	0.25
-1	-10	0	-7	0	0	-8	-5	2	-1	0	0	0	0	-2	1	0	0	0	0	6
1	2	2	-3	0	5	-1	-1	1	0	-1	0	0	0	-1	0	1	0	0	0	4
0	-10	1	-7	0	-2	-5	-13	2	0	0	-1	0	0	-2	0	0	1	0	0	5
-1	3	1	0	0	-1	2	-4	1	0	0	0	-1	0	-1	0	0	0	1	0	8
-2.75	-4.5	2.25	-3.25	0	-0.5	-5.25	-6.75	1.25	0	0	0	0	-1	-1.25	0	0	0	0	1	1.75

Разрешающий столбец 9, разрешающая строка 6, разрешающий элемент 1.25.

...

Девятая итерация:

0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0
0	0.52	1.41	0.6	1	0.36	0	0	0	0.06	0	-0.008	0.03	-0.31	0	-0.06	0	0.008	-0.03	0.31	0.21
1	-1.48	-0.73	-0.78	0	-0.08	0	0	0	-0.1	0	-0.18	0.11	0.36	0	0.1	0	0.18	-0.16	-0.36	0.004
0	0.3	-0.2	0.12	0	0.2	0	1	0	-0.12	0	0.11	-0.05	0.06	0	0.12	0	-0.11	0.05	-0.06	0.48
0	1.28	0.06	0.55	0	-0.12	1	0	0	0.05	0	0.01	-0.17	0.05	0.0	-0.05	0	-0.01	0.17	-0.05	0.95
0	-4.92	-3.22	0.13	0	-5.2	0	0	0	-0.69	1	-0.08	-0.43	0.77	0	0.69	-1	0.08	0.43	-0.77	2.6
0	0.14	-0.63	-1.44	0	-0.06	0	0	1	-0.67	0	0.22	-0.76	0.52	-1	0.67	0	-0.22	0.76	-0.52	8.03

Начальное решение для двойственной задачи:

$$\begin{pmatrix}0.004 & 0 & 0 & 0 & 0.21 & 0 & 0.95 & 0.48 & 8.03 & 0 & 2.6 & 0 & 0 & 0\end{pmatrix}$$

Для нахождения решения двойственной задачи продолжим с найденной угловой точки. Исключим столбцы, соответствующие элементам u и заменим нулевую строку на $(b, 0)$

Построим симплекс таблицу для двойственной задачи:

3	6	5	2	7	2	1	4	0	0	0	0	0	0	0
0	0.52	1.41	0.6	1	0.36	0	0	0	0.06	0	-0.008	0.03	-0.31	0.21
1	-1.48	-0.73	-0.78	0	-0.08	0	0	0	-0.1	0	-0.18	0.11	0.36	0.004
0	0.3	-0.2	0.12	0	0.2	0	1	0	-0.12	0	0.11	-0.05	0.06	0.48
0	1.28	0.06	0.55	0	-0.12	1	0	0	0.05	0	0.01	-0.17	0.05	0.95
0	-4.92	-3.22	0.13	0	-5.2	0	0	0	-0.69	1	-0.08	-0.43	0.77	2.6
0	0.14	-0.63	-1.44	0	-0.06	0	0	1	-0.67	0	0.22	-0.76	0.52	8.03

Для запуска симплекс-метода нужно преобразовать ее к виду с базисными столбцами используя элементарные преобразования.

0	4.33	-1.97	-0.87	0	-0.95	0	0	0	0.31	0	0.17	-0.18	0.77	-4.4
0	0.52	1.41	0.6	1	0.36	0	0	0	0.06	0	-0.008	0.03	-0.31	0.21
1	-1.48	-0.73	-0.78	0	-0.08	0	0	0	-0.1	0	-0.18	0.11	0.36	0.004
0	0.3	-0.2	0.12	0	0.2	0	1	0	-0.12	0	0.11	-0.05	0.06	0.48
0	1.28	0.06	0.55	0	-0.12	1	0	0	0.05	0	0.01	-0.17	0.05	0.95
0	-4.92	-3.22	0.13	0	-5.2	0	0	0	-0.69	1	-0.08	-0.43	0.77	2.6
0	0.14	-0.63	-1.44	0	-0.06	0	0	1	-0.67	0	0.22	-0.76	0.52	8.03

Первая итерация:

0	4.33	-1.97	-0.87	0	-0.95	0	0	0	0.31	0	0.17	-0.18	0.77	-4.4
0	0.52	1.41	0.6	1	0.36	0	0	0	0.06	0	-0.008	0.03	-0.31	0.21
1	-1.48	-0.73	-0.78	0	-0.08	0	0	0	-0.1	0	-0.18	0.11	0.36	0.004
0	0.3	-0.2	0.12	0	0.2	0	1	0	-0.12	0	0.11	-0.05	0.06	0.48
0	1.28	0.06	0.55	0	-0.12	1	0	0	0.05	0	0.01	-0.17	0.05	0.95
0	-4.92	-3.22	0.13	0	-5.2	0	0	0	-0.69	1	-0.08	-0.43	0.77	2.6
0	0.14	-0.63	-1.44	0	-0.06	0	0	1	-0.67	0	0.22	-0.76	0.52	8.03

Разрешающий столбец 3, разрешающая строка 1, разрешающий элемент 1.41.

Вторая итерация:

0	5.05	0	-0.03	1.4	-0.45	0	0	0	0.4	0	0.16	-0.13	0.34	-4.11
0	0.37	1	0.43	0.71	0.26	0	0	0	0.04	0	-0.006	0.02	-0.22	0.15
1	-1.21	0	-0.46	0.52	0.11	0	0	0	-0.07	0	-0.19	0.12	0.2	0.12
0	0.37	0	0.19	0.14	0.24	0	1	0	-0.11	0	0.11	-0.05	0.02	0.51
0	1.26	0	0.52	-0.04	-0.14	1	0	0	0.04	0	0.01	-0.17	0.06	0.95
0	-3.74	0	1.51	2.28	-4.38	0	0	0	-0.55	1	-0.1	-0.35	0.07	3.08
0	0.37	0	-1.17	0.44	0.11	0	0	1	-0.64	0	0.22	-0.74	0.38	8.12

Разрешающий столбец 6, разрешающая строка 1, разрешающий элемент 0.26.

Третья итерация:

0	5.69	1.74	0.72	2.63	0	0	0	0	0.47	0	0.15	-0.09	-0.03	-3.84
0	1.43	3.89	1.66	2.75	1	0	0	0	0.16	0	-0.02	0.09	-0.84	0.59
1	-1.37	-0.42	-0.65	0.22	0	0	0	0	-0.08	0	-0.19	0.11	0.29	0.05
0	0.02	-0.95	-0.21	-0.53	0	0	1	0	-0.15	0	0.11	-0.07	0.22	0.37
0	1.46	0.54	0.75	0.34	0	1	0	0	0.07	0	0.01	-0.16	-0.06	1.03
0	2.53	17.01	8.77	14.31	0	0	0	0	0.14	1	-0.2	0.05	-3.61	5.67
0	0.22	-0.41	-1.34	0.16	0	0	0	1	-0.66	0	0.22	-0.75	0.47	8.06

Разрешающий столбец 13, разрешающая строка 2, разрешающий элемент 0.11.

Четвертая итерация:

0.8	4.6	1.4	0.2	2.8	0	0	0	0	0.4	0	0	0	0.2	-3.8
-0.8	2.5	4.23	2.18	2.58	1	0	0	0	0.22	0	0.13	0	-1.075	0.55
8.8	-12.03	-3.73	-5.68	1.93	0	0	0	0	-0.73	0	-1.63	1	2.58	0.45
0.6	-0.8	-1.2	-0.6	-0.4	0	0	1	0	-0.2	0	0	0	0.4	0.4
1.4	-0.45	-0.05	-0.15	0.65	0	1	0	0	-0.05	0	-0.25	0	0.35	1.1
-0.4	3.08	17.18	9.03	14.23	0	0	0	0	0.18	1	-0.13	0	-3.73	5.65
6.6	-8.8	-3.2	-5.6	1.6	0	0	0	1	-1.2	0	-1	0.0	2.4	8.4

Отрицательные элементы в первой строке отсутсвуют. Оптимальное решение:

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0.55 & 1.1 & 0.4 \end{pmatrix}$$

Целевая функция равна 3.8

```
from prettytable import PrettyTable

def simplex(cons, z, b):
        table_output(cons, b, z, 'Начальная симплекс таблица:')

        basis = [0] * len(z)
        for i in range(6, 14):
                basis[i] = True

        step = 0
        while True:
                step += 1

                sign = 1
                for i in range(len(z)):
                        if z[i] < 0:
                                sign = -1
                if sign > 0:
                        break

                min_number = z[0]
                min_column = 0
                for i in range(len(z)):
                        if z[i] < min_number:
                                min_number = z[i]
                                min_column = i

                min_row = -1
                for i in range(len(cons)):
                        xi = cons[i][min_column]
                        if xi == 0:
                                continue
                        if b[i] / xi < 0:
                                continue
                        if min_row == -1:
                                b_min = b[i] / xi
                                min_row = i
                        else:
                                if (b[i] / xi) < b_min:
                                        b_min = b[i] / xi
                                        min_row = i

                basis[min_column] = True
                for i in range(len(cons[min_row])):
                        if cons[min_row][i] == 1 and basis[i] == True
                         and i != min_column:
                                basis[i] = 0

                divider = cons[min_row][min_column]
                for i in range(len(cons[min_row])):
                        cons[min_row][i] = cons[min_row][i] / divider
                b[min_row] = b[min_row] / divider

                for i in range(len(cons)):
                        if i == min_row:
                                continue
                        factor = cons[i][min_column] /
                         cons[min_row][min_column]
```

```
            for j in range(len(cons[min_row])):
                a = cons[min_row][j] * factor
                cons[i][j] = cons[i][j] - a
            b[i] = b[i] - b[min_row] * factor
        factor = z[min_column] / cons[min_row][min_column]
        for i in range(len(z)):
            z[i] = z[i] - cons[min_row][i] * factor
        b[len(b) - 1] = b[len(b) - 1] - b[min_row] * factor
        table_output(cons, b, z, 'Шаг {}:'.format(step))

        print('Текущий базис:', end='\t')
        for i in range(len(basis)):
            if basis[i]:
                print(letters[i], end=' ')
        print()
    table_output(cons, b, z, 'Конечная симплекс таблица:')
    for i in range(len(basis)):
        if basis[i]:
            print(letters[i], end=' ')
    print()
    for i in range(len(cons)):
        for j in range(len(cons[min_row])):
            if cons[i][j] == 1 and basis[j] == True:
                basis[j] = b[i]
    for i in range(len(basis)):
        if basis[i]:
            print(letters[i], '=', basis[i])
    print(basis)
    print('\n')
```

```

def dual(cons, z, b):

    table_output(cons, b, z, 'Начальная симплекс таблица:')

    basis = [False] * len(z)
    basis[0] = True
    basis[2] = True
    basis[3] = True
    basis[4] = True
    basis[5] = True
    basis[6] = True
    basis[9] = True
    basis[13] = True
    step = 0
    while True:
        step += 1
        sign = 1
        for i in range(len(z)):
            if round(z[i], 2) < 0:
                sign = -1
        if sign > 0:
            break

        min_number = z[0]
        min_column = 0
```

```
            for i in range(len(z)):
                if z[i] < min_number:
                    min_number = z[i]
                    min_column = i

            min_row = -1
            for i in range(len(cons)):
                xi = cons[i][min_column]
                if xi == 0:
                    continue
                if xi < 0:
                    continue
                if min_row == -1:
                    b_min = b[i] / xi
                    min_row = i
                else:
                    if (b[i] / xi) < b_min:
                        b_min = b[i] / xi
                        min_row = i

            basis[min_column] = True
            for i in range(len(cons[min_row])):
                if cons[min_row][i] == 1 and basis[i] == True and
                 i != min_column:
                    basis[i] = False


            divider = cons[min_row][min_column]
            for i in range(len(cons[min_row])):
                cons[min_row][i] = cons[min_row][i] / divider
            b[min_row] = b[min_row] / divider

            for i in range(len(cons)):
                if i == min_row:
                    continue
                factor = cons[i][min_column] /
                 cons[min_row][min_column]
                for j in range(len(cons[min_row])):
                    a = cons[min_row][j] * factor
                    cons[i][j] = cons[i][j] - a
                b[i] = b[i] - b[min_row] * factor
            factor = z[min_column] / cons[min_row][min_column]
            for i in range(len(z)):
                z[i] = z[i] - cons[min_row][i] * factor
            b[len(b) - 1] = b[len(b) - 1] - b[min_row] * factor
            table_output(cons, b, z, 'Шаг {}:'.format(step))

        table_output(cons, b, z, 'Конечная симплекс таблица:')
        print('Конечный базис:', end='\t')

        print()
        for i in range(len(cons)):
            for j in range(len(cons[min_row])):
                if cons[i][j] == 1 and basis[j] == True:
                    basis[j] = b[i]
        print('\Ответn: ', end="")
        print(basis)
        print('\n')
```