МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГБОУ ВО АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Институт цифровых технологий, электроники и физики

Кафедра вычислительной техники и электроники

Отчёт по лабораторным работам. 13 вариант

(по курсу «Методы оптимизации»)

Выполнил: ст. 595 гр.:

___________ Д. В. Осипенко

Проверил: к.ф-м. наук, доцент каф. ВТиЭ

___________ В. И. Иордан

«___» _________ 2022 г.

Барнаул, 2022 г.

Содержание

1 Графический метод решения задач линейного программирования

1.1 Краткие теоретические сведения по теме лабораторной работы

Пусть необходимо найти максимальное значение функции $Z(x) = c_1x_1 + c_2x_2$ при ограничениях:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 \le b_1 \\ a_{21}x_1 + a_{22}x_2 \le b_2 \\ \ldots \\ a_{m1}x_1 + a_{m2}x_2 \le b_m \end{cases} \qquad x_1 \ge 0, x_2 \ge 0$$

Допустим, что система ограничений имеет решение, а ее многоугольник решений ограничен.

Каждое из неравенств ограничений определяет полуплоскость с границей $a_{i1} + x_1 + ai2x_2 = b_i, i = 1 \ldots m$ или $x_1 = 0, x_2 = 0$

Так как множество точек пересечения данных полуплоскостей - выпуклое, то областью допустимых решений задачи является выпуклое множество, которое называется многоугольником решений. Стороны этого многоугольника лежат на прямых, уравнения которых получаются из исходной системы ограничений заменой знаков неравенств на знаки точных равенств.

Представим этот многоугольник на плоскости:

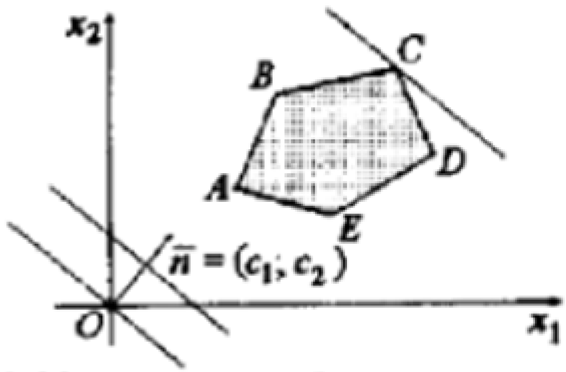


Рис. 1.1. Многоугольник области допустимых решений

Линейная функция при фиксированных значениях $Z(x) = 0$ является уравнением прямой линии $c_1x_1 + c_2x_2 = const$. Прямая, соответствующая данной функции, проходит через начало координат. Другим значениям соответствует прямые параллельные друг другу.

Прямая, уравнение которой получается из целевой функции, если ее приравнять постоянной величине, называется линией уровня.

Вектор нормали уровня $\vec{n}$ имеет координаты c1 и c2.

Если перемещать линию уровня параллельно своему начальному положению в направлении вектора $\vec{n}$, то последней точкой, в которой линия уровня коснется области допустимых решений (ОДР) будет точка c.

Линия уровня, имеющая общие точки с ОДР и расположенная так, что ОДР целиком находится в одной из полуплоскостей называется опорной прямой.

Алгоритм решения задачи линейного программирования:

1. Строится область допустимых решений.
2. Строится вектор $\vec{n} = (c_1, c_2)$ с точкой приложения в начале координат.
3. Перпендикулярно вектору $\vec{n}$ проводится одна из линий уровня
4. Линия уровня перемещается параллельно самой себе до положения опорной прямой. На этой прямой находится максимум или минимум функции

1.2 Решение индивидуального задания

Дана задача:

$$Z(X) = 2x_1 + 4x_2 \to \min$$
$$\begin{cases} 2x_1 - x_2 \geq 0 \\ -x_1 - x_2 \leq 0 \\ 3x_1 + 7x_2 \leq 40 \\ 8x_1 - 4x_2 \leq 26 \end{cases}$$
$$x_1 \geq 0, \quad x_2 \geq 0$$

Изобразим на плоскости систему координат Ox_1x_2 и построим граничные прямые ОДР:

$$x_1 \geq 0, \quad x_2 \geq 0$$
$$\begin{cases} 2x_1 - x_2 \geq 0, (1) \\ -x_1 - x_2 \leq 0, (2) \\ 3x_1 + 7x_2 \leq 40, (3) \\ 8x_1 - 4x_2 \leq 26, (4) \end{cases}$$

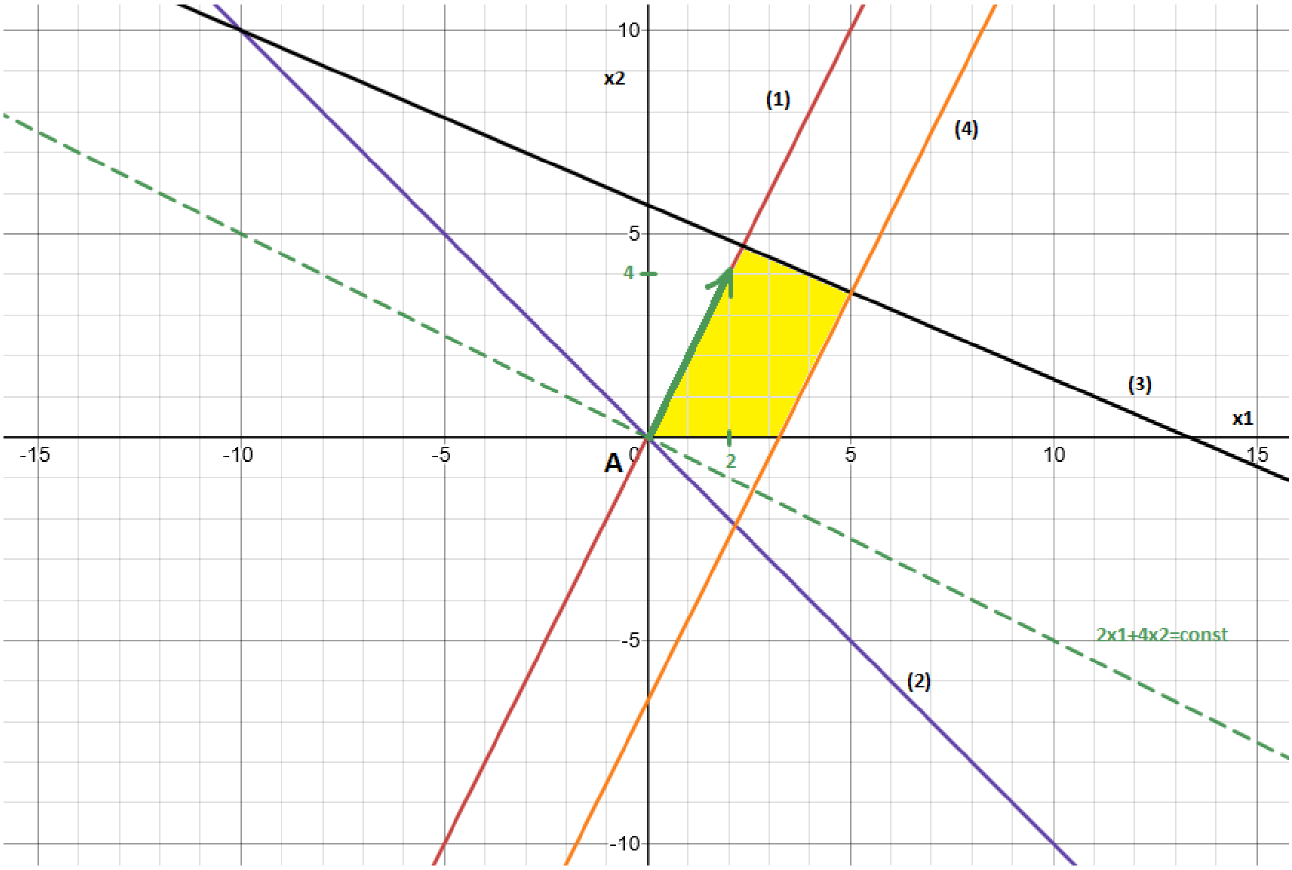


Рис. 1.2 Область допустимых решений

Для линий уровня $2x_1 + 4x_2 = \text{const}$ строим нормальный вектор $\vec{n} = (2, 4)$ перпендикулярно вектору нормаль построим одну из линий уровня Перемещаем её в направлении вектора $\vec{n}$ до опорной прямой. Для определения координат точки А решаем систему уравнений:

$$\begin{cases} 2x_1 - x_2 \geq 0, (1) \\ -x_1 - x_2 \leq 0, (2) \end{cases}$$

Получаем $x_1 = 0, x_2 = 0$ это и есть оптимальное решение. Минимальное значение целевой функции $Z(X) = 2 \cdot 0 + 4 \cdot 0 = 0$.

2 Транспортная задача

2.1 Краткие теоретические сведения по теме лабораторной работы

Транспортная задача (ТЗ) — одна из распространенных задач линейного программирования. Ее цель — разработка наиболее рациональных путей и способов транспортирования товаров, устранение чрезмерно дальних, встречных, повторных перевозок.

Задана матрица $c = (c_{ij})$ транспортных расходов: затраты на перевозку единицы продукции из пункта производства i в пункт потребления j.

Требуется составить план перевозок, который не выводит за пределы мощностей производителей, удовлетворяет полностью всех потребителей и минимизирует суммарные затраты на перевозки.

Постановка задачи: Пусть имеется m пунктов производства и n пунктов потребления одного продукта. Объем производства в пункте производства с номером i равен $_i$, объём потребления в пункте потребления с номером j равен $b_j, (i = 1, 2, \ldots, m; j = 1, 2, \ldots n)$.

Введем обозначение: $_{ij}$ — количество груза, которое нужно перевезти из i-го пункта отправления в j-й пункт назначения. Так как нужно перевезти весь груз из каждого пункта отправления $_i$, то должны выполняться равенства:

$$\begin{cases} x_{11} + x_{12} + \ldots + x_{1n} = a_1 \\ \ldots \\ x_{m1} + x_{m2} + \ldots + x_{mn} = a_m \end{cases}$$

Размер поставок должен выражаться неотрицательным числом: $x_{ij} \geq 0, i = \overline{1, m}, j = \overline{1, n}$. Стоимость всех запланированный перевозок должна быть минимальной:

$$F(x) = c_{11}x_{11} + \ldots + c_{mn}x_{mn} \to \min$$

$$\begin{cases} x_{11} + \ldots + x_{m1} = b_1 \\ \ldots \\ x_{1n} + \ldots + x_{mn} = b_n \end{cases}$$

В рассмотренной модели ТЗ предполагается, что суммарные запасы поставщиков равны суммарным запросам потребителей. Такая задача называется задачей с правильным балансом (сбалансированной задачей), ее модель — закрытой. В противном случае транспортная задача линейного программирования называется открытой.

2.2 Решение индивидуального задания

2.2.1 exel

Дана задача

Вариант №13

Производители	Потребители				Объем производства
	B_1	B_2	B_3	B_4	
A_1	15	3	23	6	70
A_2	1	4	17	8	47
A_3	9	13	14	7	38
Спрос	20	30	40	50	

Создаем таблицу для ввода условий задачи и введем исходные данные:

Матрица стоимости перевозок C				
	Потребитель B1	Потребитель B2	Потребитель B3	Фиктивный потребитель B4
Склад A1	15	3	23	6
Склад A2	1	4	17	8
Склад A3	9	13	14	7

Матрица перевозок X						
	Потребитель B1	Потребитель B2	Потребитель B3	Фиктивный потребитель B4	Доставлено	Запасы
Склад A1	0	0	0	0	0	70
Склад A2	0	0	0	0	0	47
Склад A3	0	0	0	0	0	38
Вывезено	0	0	0	0		
Потребности потребителей	20	30	40	50		

Целевая функция	0

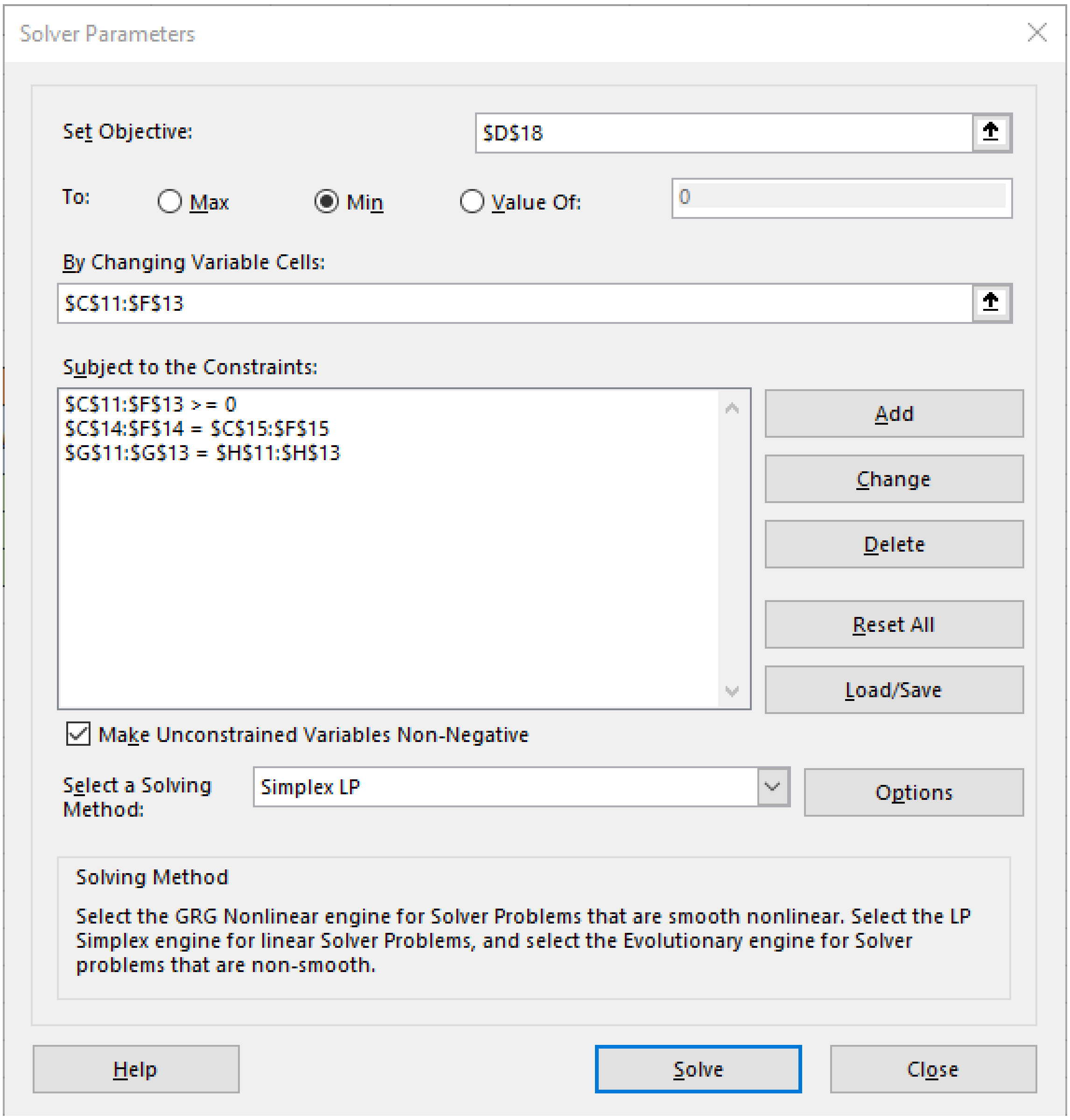
Вводим формулы расчета для различных ячеек:

```
D18 =SUMPRODUCT(C4:F6;C11:F13)

G11 =SUM(C11:F11)
G12 =SUM(C12:F12)
G13 =SUM(C13:F13)

C14 =SUM(C11:C13)
D14 =SUM(D11:D13)
E14 =SUM(E11:E13)
```

```
F14 =SUM(F11:F13)
```

	A	B	C	D	E	F	G	H
1								
2		Матрица стоимости перевозок C						
3			Потребитель B1	Потребитель B2	Потребитель B3	Фиктивный потребитель B4		
4		Склад A1	15	3	23	6		
5		Склад A2	1	4	17	8		
6		Склад A3	9	13	14	7		
7								
8								
9		Матрица перевозок X						
10			Потребитель B1	Потребитель B2	Потребитель B3	Фиктивный потребитель B4	Доставлено	Запасы
11		Склад A1	0	0	0	0	=SUM(C11:F11)	70
12		Склад A2	0	0	0	0	=SUM(C12:F12)	47
13		Склад A3	0	0	0	0	=SUM(C13:F13)	38
14		Вывезено	=SUM(C11:C13)	=SUM(D11:D13)	=SUM(E11:E13)	=SUM(F11:F13)		
15		Потребности потребителей	20	30	40	50		
16								
17								
18		Целевая функция		=SUMPRODUCT(C4:F6;C11:F13)				
19								

Заполняем окно параметров поиска решений:



Результат:

	Матрица перевозок X						
	Потребитель B1	Потребитель B2	Потребитель B3	Фиктивный потребитель B4	Доставлено	Запасы
Склад A1	0	30	40	0	70	70
Склад A2	20	0	0	12	32	47
Склад A3	0	0	0	38	38	38
Вывезено	20	30	40	50		
Потребности потребителей	20	30	40	50		
Целевая функция	1392					

2.2.2 Метод потенциалов(опорный план с помощью северо-западного угла)

Задана таблица транспортной задачи:

	B1=20	B2=30	B3=40	B4=50
A1 = 70	15	3	23	6
A2 = 47	1	4	17	8
A3 = 38	9	13	14	7

Суммарные запасы груза $70 + 47 + 38 = 155$, а суммарное потребление $20 + 30 + 40 + 50 = 140$. Следовательно Задача является открытого типа и ее нужно закрыть вводом нового потребителя с стоимостью перевозок 0 и потребностями $155 - 140 = 15$.

	B1=20	B2=30	B3=40	B4=50	B5=15
A1 = 70	15	3	23	6	0
A2 = 47	1	4	17	8	0
A3 = 38	9	13	14	7	0

Метод северо-западного угла:

	B1	B2	B3	B4	B5	Запасы
A1	15[20]	3[30]	23[20]	6[0]	0[0]	70
A2	1[0]	4[0]	17[20]	8[27]	0[0]	47
A3	9[0]	13[0]	14[0]	7[23]	0[15]	38
Потребности	20	30	40	50	15	

$7 = m + n - 1 = 3 + 5 - 1 = 7 \Rightarrow$ невырожденный
$F(x) = \sum_{i=1}^{3} \sum_{j=1}^{5} c_{ij}x_{ij} = 15 \cdot 20 + 3 \cdot 30 + 23 \cdot 20 + 17 \cdot 20 + 8 \cdot 27 + 7 \cdot 23 + 0 \cdot 15 = 1567$

Метод потенциалов:

1. Находим предварительные потенциалы u_i, v_j, по заданному плану, где $u_i + v_j = c_{ij}, u_1 = 0$
2. Проверяем на оптимальность, где не существуют $u_i + v_j > c_{ij}$
3. Выбераем максимальную оценку свободной клетки
4. Строим цикл, чередуя +/-, вершина цикла, выбранная свободная клетка начинается с '+', выбераем наименьшей объем груза из ячеек с '-', и прибавляем это значение к элементам цикла

1-итерация:

1)
$$\begin{aligned}
u_1+v_1&=15; v_1=15-0=15\\
u_1+v_2&=3; v_2=3-0=3\\
u_1+v_3&=23; v_3=23-0=23\\
u_2+v_3&=17; u_2=17-23=-6\\
u_2+v_4&=8; v_4=8-(-6)=14\\
u_3+v_4&=7; u_3=7-14=-7\\
u_3+v_5&=0; v_5=0-(-7)=7
\end{aligned}$$

2)
$$\begin{aligned}
\Delta_{14}&=0+14-6=8>0\\
\Delta_{15}&=0+7-0=7>0\\
\Delta_{21}&=-6+15-1=8>0\\
\Delta_{22}&=-6+3-4=-7<0\\
\Delta_{25}&=-6+7-0=1>0\\
\Delta_{31}&=-7+15-9=-1<0\\
\Delta_{32}&=-7+3-13=-17<0\\
\Delta_{33}&=-7+23-14=2>0
\end{aligned}$$

3) $\max(8,7,8,1,2)=8 \Rightarrow \max(1_{21}, 6_{14})=6$

	B1(V1=15)	B2(V2=3)	B3(V3=23)	B4(V4=14)	B5(V5=7)	Запасы
A1(U1=0)	15[20]	3[30]	23[20]-	6[0]+	0[0]	70
A2(U2=-6)	1[0]	4[0]	17[20]+	8[27]-	0[0]	47
A3(U3=-7)	9[0]	13[0]	14[0]	7[23]	0[15]	38
Потребности	20	30	40	50	15	

$$\Downarrow$$

4) $(1,4)\to(1,3)\to(2,3)\to(2,4)\to(1,4)$

$$\min(20,27)=20$$

	B1(V1=15)	B2(V2=3)	B3(V3=23)	B4(V4=14)	B5(V5=7)	Запасы
A1(U1=0)	15[20]	3[30]	23[0]	6[20]	0[0]	70
A2(U2=-6)	1[0]	4[0]	17[40]	8[07]	0[0]	47
A3(U3=-7)	9[0]	13[0]	14[0]	7[23]	0[15]	38
Потребности	20	30	40	50	15	

2-итерация:

1)
$$\begin{aligned}
u_1+v_1&=15; v_1=15-0=15\\
u_1+v_2&=3; v_2=3-0=3\\
u_1+v_4&=6; v_4=6-0=6\\
u_2+v_4&=8; u_2=8-6=2\\
u_2+v_3&=17; v_3=17-2=15\\
u_3+v_4&=7; u_3=7-6=1\\
u_3+v_5&=0; v_5=0-1=-1
\end{aligned}$$

2)
$$\begin{aligned}
\Delta_{13}&=0+15-23=-8<0\\
\Delta_{15}&=0+(-1)-0=-1<0\\
\Delta_{21}&=2+15-1=16>0\\
\Delta_{22}&=2+3-4=-1<0\\
\Delta_{25}&=2+-1-0=1>0\\
\Delta_{31}&=1+15-9=7>0\\
\Delta_{32}&=1+3-13=-9<0\\
\Delta_{33}&=1+15-14=2>0
\end{aligned}$$

3) $\max(16,1,7,2)=16 \Rightarrow \max(16_{21})=1$

	B1(V1=15)	B2(V2=3)	B3(V3=15)	B4(V4=6)	B5(V5=-1)	Запасы
A1(U1=0)	15[20]-	3[30]	23[20]	6[0]+	0[0]	70
A2(U2=2)	1[0]+	4[0]	17[40]	8[7]-	0[0]	47
A3(U3=1)	9[0]	13[0]	14[0]	7[23]	0[15]	38
Потребности	20	30	40	50	15	

$$\Downarrow$$

4) $(2,1) \to (1,1) \to (1,4) \to (2,4) \to (2,1)$

$$\min(20,7) = 7$$

	B1	B2	B3)	B4	B5	Запасы
A1	15[13]	3[30]	23[0]	6[27]	0[0]	70
A2	1[7]	4[0]	17[40]	8[0]	0[0]	47
A3	9[0]	13[0]	14[0]	7[23]	0[15]	38
Потребности	20	30	40	50	15	

3-итерация:

1)
$$\begin{aligned} u_1 + v_1 &= 15; v_1 = 15 - 0 = 15 \\ u_2 + v_1 &= 1; u_2 = 1 - 15 = -14 \\ u_2 + v_3 &= 17; v_3 = 17 + 14 = 31 \\ u_1 + v_2 &= 3; v_2 = 3 - 0 = 3 \\ u_1 + v_4 &= 6; v_4 = 6 - 0 = 6 \\ u_3 + v_4 &= 7; u_3 = 7 - 6 = 1 \\ u_3 + v_5 &= 0; v_5 = 0 - 1 = -1 \end{aligned}$$

2)
$$\begin{aligned} \Delta_{13} &= 0 + 31 - 23 = 8 > 0 \\ \Delta_{31} &= 1 + 15 - 9 = 7 > 0 \\ \Delta_{33} &= 1 + 31 - 14 = 18 > 0 \end{aligned}$$

3) $\max(8,7,18) = 18 \Rightarrow \max(18_{33}) = 14$

	B1(V1=15)	B2(V2=3)	B3(V3=31)	B4(V4=6)	B5(V5=-1)	Запасы
A1(U1=0)	15[13]-	3[30]	23[0]	6[27]+	0[0]	70
A2(U2=-14)	1[7]+	4[0]	17[40]-	8[0]	0[0]	47
A3(U3=1)	9[0]	13[0]	14[0]+	7[23]-	0[15]	38
Потребности	20	30	40	50	15	

$$\Downarrow$$

4) $(3,3) \to (2,3) \to (2,1) \to (1,1) \to (1,4) \to (3,4) \to (3,3)$

$$\min(13,23,40) = 13$$

	B1	B2	B3	B4	B5	Запасы
A1	15[0]	3[30]	23[0]	6[40]	0[0]	70
A2	1[20]	4[0]	17[27]	8[0]	0[0]	47
A3	9[0]	13[0]	14[13]	7[10]	0[15]	38
Потребности	20	30	40	50	15	

4-итерация:

1)
$$\begin{aligned} v_2 &= 3 - u_1 = 3 - 0 = 3 \\ v_4 &= 6 - u_1 = 6 - 0 = 6 \\ u_3 &= 7 - v_4 = 7 - 6 = 1 \\ v_3 &= 14 - u_3 = 14 - 1 = 13 \\ u_2 &= 17 - v_3 = 17 - 13 = 4 \\ v_1 &= 1 - u_2 = 1 - 4 = -3 \\ v_5 &= 0 - u_3 = 0 - 1 = -1 \end{aligned}$$

2)
$$\begin{aligned} \Delta_{22} &= 4 + 3 - 4 = 3 > 0 \\ \Delta_{24} &= 4 + 6 - 8 = 2 > 0 \\ \Delta_{25} &= 4 - 1 - 0 = 3 > 0 \end{aligned}$$

3) $\max(3, 2, 3) = 3 \Rightarrow \max(3_{22}) = 4$

	B1(V1=-3)	B2(V2=3)	B3(V3=13)	B4(V4=6)	B5(V5=-1)	Запасы
A1(U1=0)	15[0]	3[30]-	23[0]	6[40]+	0[0]	70
A2(U2=4)	1[20]	4[0]+	17[27]-	8[0]	0[0]	47
A3(U3=1)	9[0]	13[0]	14[13]+	7[10]-	0[15]	38
Потребности	20	30	40	50	15	

$$\Downarrow$$

4) $(2, 2) \to (1, 2) \to (1, 4) \to (3, 4) \to (3, 3) \to (2, 3) \to (2, 2)$

$$\min(30, 10, 27) = 10$$

	B1	B2	B3	B4	B5	Запасы
A1	15[0]	3[20]	23[0]	6[50]	0[0]	70
A2	1[20]	4[10]	17[17]	8[0]	0[0]	47
A3	9[0]	13[0]	14[23]	7[0]	0[15]	38
Потребности	20	30	40	50	15	

5-итерация:

1)
$$\begin{aligned} v_2 &= 3 - u_1 = 3 - 0 = 3 \\ u_2 &= 4 - v_2 = 4 - 3 = 1 \\ v_1 &= 1 - u_2 = 1 - 1 = 0 \\ v_3 &= 17 - u_2 = 17 - 1 = 16 \\ u_3 &= 14 - v_3 = 14 - 16 = -2 \\ v_5 &= 0 - u_3 = 0 + 2 = 2 \\ v_4 &= 6 - u_1 = 6 - 0 = 6 \end{aligned}$$

2)
$$\begin{aligned} \Delta_{15} &= 0 + 2 - 0 = 2 > 0 \\ \Delta_{25} &= 1 + 2 - 0 = 3 > 0 \end{aligned}$$

3) $\max(2, 3) = 3 \Rightarrow \max(3_{25}) = 0$

	B1(V1=0)	B2(V2=3)	B3(V3=16)	B4(V4=6)	B5(V5=2)	Запасы
A1(U1=0)	15[0]	3[20]	23[0]	6[50]	0[0]	70
A2(U2=1)	1[20]	4[10]	17[17]-	8[0]	0[0]+	47
A3(U3=-2)	9[0]	13[0]	14[23]+	7[0]	0[15]-	38
Потребности	20	30	40	50	15	

$$\Downarrow$$

4) $(2,5) \to (3,5) \to (3,3) \to (2,3) \to (2,5)$

$$\min(17,15) = 15$$

	B1	B2	B3	B4	B5	Запасы
A1	15[0]	3[20]	23[0]	6[50]	0[0]	70
A2	1[20]	4[10]	17[2]	8[0]	0[15]	47
A3	9[0]	13[0]	14[38]	7[0]	0[0]	38
Потребности	20	30	40	50	15	

6-итерация:

1) $$\begin{aligned} v_2 &= 3 - u_1 = 3 - 0 = 3 \\ u_2 &= 4 - v_2 = 4 - 3 = 1 \\ v_1 &= 1 - u_2 = 1 - 1 = 0 \\ v_3 &= 17 - u_2 = 17 - 1 = 16 \\ u_3 &= 14 - v_3 = 14 - 16 = -2 \\ v_5 &= 0 - u_2 = 0 - 1 = -1 \\ v_4 &= 6 - u_1 = 6 - 0 = 6 \end{aligned}$$

2) $$\begin{aligned} \Delta_{11} &= 0 + 0 - 15 \leq 0 \\ \Delta_{13} &= 0 + 16 - 23 = -7 \leq 0 \\ \Delta_{15} &= 0 - 1 - 0 = -1 \leq 0 \\ \Delta_{24} &= 1 + 6 - 8 = -1 \leq 0 \\ \Delta_{31} &= -2 + 0 - 0 = -11 \leq 0 \\ \Delta_{32} &= -2 + 3 - 13 = -12 \leq 0 \\ \Delta_{34} &= -2 + 6 - 7 = -3 \leq 0 \\ \Delta_{35} &= -2 - 1 - 0 = -3 \leq 0 \end{aligned}$$

	B1	B2	B3	B4	B5	Запасы
A1	15[0]	3[20]	23[0]	6[50]	0[0]	70
A2	1[20]	4[10]	17[2]	8[0]	0[15]	47
A3	9[0]	13[0]	14[38]	7[0]	0[0]	38
Потребности	20	30	40	50	15	

Опорный план является оптимальным, т.к. все оценки свободных клеток удовлетворяют условию $u_i + v_j \leq c_{ij}$. Затраты: $F(x) = 3 \cdot 20 + 6 \cdot 50 + 1 \cdot 20 + 4 \cdot 10 + 17 \cdot 2 + 0 \cdot 15 + 14 \cdot 38 = 986$

3 Задача коммивояжера

3.1 Краткие теоретические сведения по теме лабораторной работы

Имеется n городов. Расстояния между любой парой городов i и j известны и составляют c ij . Коммивояжер выезжает из какого-либо города и должен посетить все города, побывав в каждом только один раз и вернуться в исходный город. Ставится задача определить такую последовательность объезда городов, или маршрут, при которой суммарная длина маршрута была бы минимальной

3.2 Решение индивидуального задания

Дана задача (12 вариант, т.к. 13 нету):

a	b	c	d	e	f	g	h	k	m	n	p	q	r	s	t	x	y	z	w
8	5	15	6	6	5	5	2	6	5	5	6	5	3	5	4	8	4	5	9

Матрица растояний (C):

	n1	n2	n3	n4	n5
n1	∞	8	5	15	6
n2	6	∞	5	5	2
n3	6	5	∞	5	6
n4	5	3	5	∞	4
n5	8	4	5	9	∞

3.3 MS Exel

Начнём работу в электронной таблице MS Excel. Создадим на листе матрицу состояний C, заполнив её исходными данными:

	B	C	D	E	F	G	H	I
1				Матрица состояния С				
2	Пункты	N1	N2	N3	N4	N5	N6	
3	N1	1000	8	5	15	6	1000	
4	N2	6	1000	5	5	2	1000	
5	N3	6	5	1000	5	6	1000	
6	N4	5	3	5	1000	4	1000	
7	N5	8	4	5	9	1000	1000	
8	N6	1000	1000	1000	1000	1000	1000	
9				Матрица переменных X				
10	Пункты	N1	N2	N3	N4	N5	N6	Выход
11	N1	0	0	0	0	0	0	=SUM(C11:H11)
12	N2	0	0	0	0	0	0	=SUM(C12:H12)
13	N3	0	0	0	0	0	0	=SUM(C13:H13)
14	N4	0	0	0	0	0	0	=SUM(C14:H14)
15	N5	0	0	0	0	0	0	=SUM(C15:H15)
16	N6	0	0	0	0	0	0	=SUM(C16:H16)
17	Вход	=SUM(C11:C16)	=SUM(D11:D16)	=SUM(E11:E16)	=SUM(F11:F16)	=SUM(G11:G16)	=SUM(H11:H16)	
18					Дополнительные переменные			
19			u2	u3	u4	u5	u6	
20								
21		Специальное условие (обеспечивает устранение нескольких несвязанных между собой маршрутов и циклов)						
22	u2+ui+n*x2i		0	=$D$20+E$20+5*E12	=$D$20+F$20+5*F12	=$D$20+G$20+5*G12	=$D$20+H$20+5*H12	
23	u3+ui+n*x3i		=$E$20+D$20+5*D13	0	=$E$20+F$20+5*F13	=$E$20+G$20+5*G13	=$E$20+H$20+5*H13	
24	u4+ui+n*x4i		=$F$20+D$20+5*D14	=$F$20+E$20+5*E14	0	=$F$20+G$20+5*G14	=$F$20+H$20+5*H14	
25	u5+ui+n*x5i		=$G$20+D$20+5*D15	=$G$20+E$20+5*E15	=$G$20+F$20+5*F15	0	=$G$20+H$20+5*H15	
26	u6+ui+n*x6i		=$H$20+D$20+5*D16	=$H$20+E$20+5*E16	=$H$20+F$20+5*F16	=$H$20+G$20+5*G16	0	
27								
28		Целевая функция		=SUMPRODUCT(C3:G7;C11:G15)				
29								

Матрицу переменных X заполняем нулями

В ячейку C16 запишем формулу: =СУММ(C11:C15). Автозаполнением скопируем эту формулу в ячейки диапазона D16:G16.

В ячейку H11 запишем формулу: =СУММ(C11:G11). Автозаполнением скопируем эту формулу в ячейки диапазона H12:15.

В ячейку F28 вводим формулу целевой функции: =СУММПРОИЗВ(C3:G7;C11:G15). В ячейки диапазона D22:G25 вводим формулы, соответствующие ограничениям:

- В ячейку E22: =$D$20-E20+6*E12. Автозаполнением копируем формулу в ячейки F26,G26;
- В ячейку D23: =$E$20-D20+6*D13. Автозаполнением копируем формулу в ячейки E23, G23;
- В ячейку D24: =$F$20-D20+6*D14. Автозаполнением копируем формулу в ячейки E24, G24;
- В ячейку D25: =$G$20-D20+6*D15. Автозаполнением копируем формулу в ячейки E25, G25.
- Ячейки D22, E23, F24, G25 = 0.

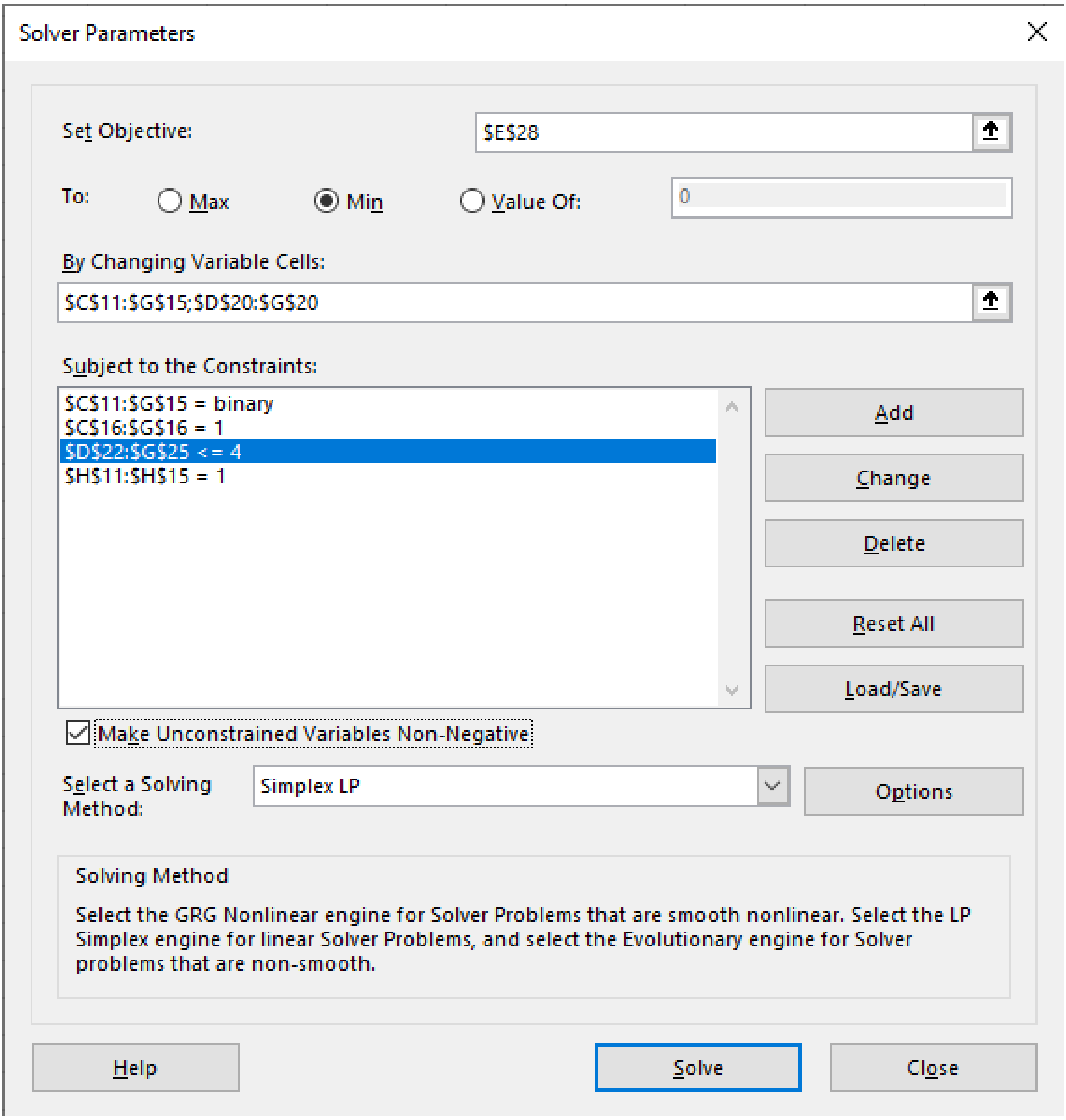
На вкладке «Данные» выбираем пункт «Поиск решения». В появившемся окне «Параметры поиска решения» (рис.2) выполняем необходимые установки:

В поле «Оптимизировать целевую функцию» вводим абсолютный адрес ячейки F28; Направление целевой функции устанавливаем «Минимум»;

В поле «Изменяя ячейки переменных» вводим абсолютный адрес диапазона ячеек \$C\$11:\$G\$15;\$D\$20:\$G\$20;

1. \$C\$11:\$G\$15 = бинарное
2. \$C\$16:\$G\$16 = 1
3. \$D\$22:\$G\$25 <= 5
4. \$H\$11:\$H\$15 = 1

Устанавливаем галочку «Сделать переменные без ограничений неотрицательными» и выбираем метод решения «Поиск решения линейных задач симплекс-методом».



Нажимаем «Найти решение». Таким образом, путь:N1-N3-N4-N2-N5-N1. Минимальная длина маршрута 23.

Матрица состояния C

Пункты	N1	N2	N3	N4	N5
N1	1000	8	5	15	6
N2	6	1000	5	5	2
N3	6	5	1000	5	6
N4	5	3	5	1000	4
N5	8	4	5	9	1000

Матрица переменных X

Пункты	N1	N2	N3	N4	N5	Выход
N1	0	0	1	0	0	1
N2	0	0	0	0	1	1
N3	0	0	0	1	0	1
N4	0	1	0	0	0	1
N5	1	0	0	0	0	1
Вход	1	1	1	1	1	

Дополнительные переменные

u2	u3	u4	u5
2	0	1	3

Специальное условие (обеспечивает устранение нескольких несвязанных между собой маршрутов и циклов)

u2-ui+n*x2i	0	2	1	4
u3-ui+n*x3i	-2	0	4	-3
u4-ui+n*x4i	4	1	0	-2
u5-ui+n*x5i	1	3	2	0

Целевая функция	23

3.4 Вывод

Задача коммивояжера может применяться для нахождения оптимального маршрута, позволяющего объехать определенные города по одному разу и вернуться в исходную точку.

4 Теория двойственности

4.1 Краткие теоретические сведения

Любой задаче линейного программирования можно поставить в соответствие другую задачу которая называется двойственной или сопряженной. Общие правила составления двойственных задач:

1. Во всех ограничениях задачи свободные члены должны находиться в правой части, а члены с неизвестными в левой.

2. Ограничения-неравенства исходной задачи должны быть записаны так, чтобы знаки неравенств были направлены в одну сторону.

3. Если знаки неравенств в исходной задаче «=<»то целевая функция должна максимизироваться, иначе минимизироваться.

4. Каждому ограничению исходной задачи соответствует неизвестное двойственной задачи.

5. Целевая функция двойственной задачи должна оптимизироваться противоположным образом по сравнению с целевой функцией исходной задачи.

4.2 Постановка задачи. 13 вариант

Составить и решить двойственную задачу и, используя ее решение, найти решение исходной задачи

$$Z(X) = x_1 + 3x_2 + \frac{2}{3}x_3 \to \min$$

$$\begin{cases} x_1 - 2x_2 + x_3 \geq 2, \\ 3x_1 + x_2 + x_3 \geq 3, \\ 2x_1 + 3x_2 - x_3 \geq 1, \end{cases} \qquad x_j \geq 0, j = 1, 2, 3$$

Составляем двойственную задачу:

$$F(Y) = -2y_1 + 3y_2 + y_3 \to \max$$

$$\begin{cases} y_1 + 3y_2 + 2y_3 \leq 1 \\ -2y_1 + y_2 + 3y_3 \leq 3 \\ y_1 + y_2 - y_3 \leq \frac{2}{3} \end{cases}$$

Введем дополнительные переменные:

$$F(Y) = -2y_1 + 3y_2 + y_3 + 0y_4 + 0y_5 + 0y_6 \to \max$$
$$\begin{cases} y_1 + 3y_2 + 2y_3 + y_4 = 1 \\ -2y_1 + y_2 + 3y_3 + y_5 = 3 \\ y_1 + y_2 - y_3 + y6 = \frac{2}{3} \end{cases}$$

Найдем решение нашей задачи с помощью симплекс метода

	Y1	Y2	Y3	Y4	Y5	Y6	Решение	Отношение
Y4	1	3	2	1	0	0	1	1/3
Y5	-2	1	3	0	1	0	3	3
Y6	1	1	-1	0	0	1	2/3	2/3
Q	2	-3	-1	0	0	0	0	

Результат 1 итерации

	Y1	Y2	Y3	Y4	Y5	Y6	Решение
Y2	1/3	1	2/3	1/3	0	0	1/3
Y5	-7/3	0	7/3	-1/3	1	0	8/3
Y6	2/3	0	-5/3	-1/3	0	1	1/3
Q	3	0	1	1	0	0	1

В строке Q отсутствуют отрицательные элементы, следовательно оптимальный план найден за 1 итерацию. Оптимальное решение двойственно задачи:

$$y_1 = 0, \quad y_2 = \frac{1}{3}, \quad y_3 = 0$$
$$Y = (0, \frac{1}{3}, 0)$$
$$\max F(Y) = -2 \cdot 0 + 3 \cdot \frac{1}{3} + 1 \cdot 0 = 1$$

Найдем оптимальное решение исходной задачи по формуле: $X^{\text{пр}}_{j\text{опт}} = -Q^{\text{дв}}_{m+j}$

$$x_1 = Q_{3+1} = Q_4 = 1 \tag{1}$$
$$x_2 = Q_{3+2} = Q_5 = 0 \tag{2}$$
$$x_3 = Q_{3+3} = Q_6 = 0 \tag{3}$$
$$X = (1; 0; 0) \tag{4}$$
$$\min Z(x) = 1 \cdot 1 + 3 \cdot 0 + \frac{2}{3} \cdot 0 = 1 \tag{5}$$

4.3 Вывод

Научились решить двойственные задачи симплекс методом. Нашли оптимальное решение двойственной и исходной задач. Решение двойственных задач применяется в экономическом анализе

5 Задача выпуклого программироваия. Теорема "Куна-Таккера"

6 Задача о рационе

6.1 Краткие теоретические сведения

Задача линейного программироваия состоит в находжении $\vec{x}$, который минимизирует целевую функцию $f^T x$, где f - вектор коэффициентов, и удовлетворяет заданным линейным ограничениям: неравенствам $Ax \leq b$ и равенствам $A_{eq}x = b_{eq}$. Кроме того, могут быть поставленны двусторонние покомпонентные ограничения в векторной форме: $lb \leq x \leq ub$.

В задачах оптимизации могут быть заданы не все типы ограничений, например, ограничения-равенства могут отсутствовать.

6.2 Решение индивидуального задания. 13 вариант.

Изветсны минимальные суточные потребности человека, в зависимости от пола и возраста, в питальных веществах и незаменимых компонентах. В табл.3 прведены содержание питательных веществ и незаменимых компонентов в 100 г. продукта. Стоимость 100 г. продуктов, включенных в диету, и предельные количества по каждому сформировать самостоятельно. Требуется рассчитать суточную диету, чтобы, с одной стороны, обеспечить минимально необходимое количество питательных веществ и незаменимых компонентов, а с другой - минимизировать стоимость разработанной диеты. При этом необходимо посчитать энергетическую ценность полученной диеты.

Список продуктов (13 вариант): *Крупа кукурузная, Хлеб ржаной из сеяной муки, Пряники заварные, Сыр костромской, Куры, Треска, Масло сливочное, Грейпфрут, Свекла, Яблоки.*

Табл 6.1 Данные к задаче

Питательные вещества	Мин. Суточная потребность, г. (для Муж. 18-25 лет)	Содержание пит. вещ. в 100 г.									
		Крупа кукурузная	Хлеб ржаной из сеяной муки	Пряники заварные	Сыр костромской	Куры	Треска	Масло сливочное	Грейпфрут	Свекла	Яблоки
Белки, г.	96	8.3	4.9	4.8	25.2	18.2	16	0.5	0.9	1.5	0.4
Жиры, г.	106	1.2	1	2.8	26.3	18.4	0.6	82.5	0.2	0.1	0.4
Углеводы, г.	420	71.6	46	77.7	0	0.7	0	0.8	6.5	9.1	9.8
Ретинол, мг.	0.19	0.2	0	0	0.17	0	0	0.38	0.02	0.01	0.03
Каротин, мг	6.6	0	0	0	0.23	0.07	0.01	0.59	0	0	0
Витамин B1, мг	1.6	0.13	0.09	0.08	0.03	0.07	0.09	0	0.05	0.02	0.03
Витамин B2, мг	1.9	0.07	0.03	0.04	0.36	0.15	0.16	0.1	0.03	0.04	0.02
Витамин PP, мг	20	1.1	0.68	0.57	0.2	7.7	2.3	0.05	0.23	0.2	0.3
Витамин C, мг	95	0	0	0	3	0	1	0	45	10	13
Стоимость 100 г., руб		1.5	4.5	1.2	1	8	1.5	3	2	6	9
Энергетическая ценность 100 г., Ккал		337	220	350	345	241	69	748	35	42	45

Решим классическую задачу линейного программирования о составлении рациона питания.

Пусть имеется 10 видов продуктов, содержащих 9 питательных веществ и незаменимых компанентов. Кроме того, известны: ежесуточная минимальная потребность организма в веществах, стоимость и энергетическая ценность (Ккал) 100 г. продукта. Требуется расчитать суточную диету так, чтобы обеспечить необходимо количество питательных веществ и незаменимых компонетов при минимальных затратах на продукты. Найти калорийность.

Требуется минимизировать затраты на приобретение продуктов. Очевидно, что количество приобретаемых продуктов не может быть отрицательным.

$$Z(X) = \frac{1}{100} \sum_{j=0}^{10} c_j x_j \to \min$$

Поскольку линейные ограничения содержат "меньше или равно а количество ингредиентов в рационе недолжно быть менее заданных величин, то следует изменить знаки обеиз частей системы

$$A = \begin{vmatrix} -8.3 & -4.9 & -4.8 & -25.2 & -18.2 & -16 & -0.5 & -0.9 & -1.5 & -0.4 \\ -1.2 & -1 & -2.8 & -26.3 & -18.4 & -0.6 & -82.5 & -0.2 & -0.1 & -0.4 \\ -71.6 & -46 & -77.7 & 0 & -0.7 & 0 & -0.8 & -6.5 & -9.1 & -9.8 \\ -0.2 & 0 & 0 & -0.17 & 0 & 0 & -0.38 & -0.02 & -0.01 & -0.03 \\ 0 & 0 & 0 & -0.23 & -0.07 & -0.01 & -0.59 & 0 & 0 & 0 \\ -0.13 & -0.09 & -0.08 & -0.03 & -0.07 & -0.09 & 0 & -0.05 & -0.02 & -0.03 \\ -0.07 & -0.03 & -0.04 & -0.36 & -0.15 & -0.16 & -0.1 & -0.03 & -0.04 & -0.02 \\ -1.1 & -0.68 & -0.57 & -0.2 & -7.7 & -2.3 & -0.05 & -0.23 & -0.2 & -0.3 \\ 0 & 0 & 0 & -3 & 0 & -1 & 0 & -45 & -10 & -13 \end{vmatrix}$$

$$b = \begin{vmatrix} -9600 \\ -10600 \\ -42000 \\ -19 \\ -660 \\ -160 \\ -190 \\ -2000 \\ -9500 \end{vmatrix}$$

```python
import numpy as np
from scipy.optimize import linprog
```

[41] 0.5s Python

```python
A = np.array([
    [8.3, 4.9, 4.8, 25.2, 18.2, 16, 0.5, 0.9, 1.5, 0.4],
    [1.2, 1, 2.8, 26.3, 18.4, 0.6, 82.5, 0.2, 0.1, 0.4],
    [71.6, 46, 77.7, 0, 0.7, 0, 0.8, 6.5, 9.1, 9.8],
    [0.2,  0, 0, 0.17, 0, 0, 0.38, 0.02, 0.01, 0.03],
    [0, 0, 0,0.23, 0.07, 0.01, 0.59, 0, 0, 0],
    [0.13, 0.09, 0.08, 0.03, 0.07, 0.09, 0, 0.05, 0.02, 0.03],
    [0.07, 0.03, 0.04, 0.36, 0.15, 0.16, 0.1, 0.03, 0.04, 0.02],
    [1.1, 0.68, 0.57, 0.2, 7.7, 2.3, 0.05, 0.23, 0.2, 0.3],
    [0, 0, 0, 3, 0, 1, 0, 45, 10, 13],
])*-1
b = np.array([
     96, 106, 420, 0.19*(10**-3), 660*(10**-3), 1.6*(10**-3), 1.9*(10**-3), 20*(10**-3), 95*(10**-3)
])*-100
bounds = (0, None)
P = np.array([
    1.5, 4.5, 1.2, 1, 8, 1.5, 3, 2, 6, 9
])
E = np.array([337, 220, 350, 345, 241, 69, 748, 35, 42,  45])

c = P*100
```

[42] 0.6s Python

```python
temp = linprog(c=c, A_ub=A, b_ub=b, bounds=bounds)
x = temp['x']
```

[43] 0.8s Python

```python
K = E.dot(x)/100
```

[44] 0.9s Python

```python
Pr = P.dot(x)/100
```

[45] 0.5s Python

Программа на Python3

$$K = 3010.7888(\textit{Ккал}), \quad Pr = 9.9093(\textit{руб})$$