МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №8

по дисциплине: Исследование операций

тема: «**Задачи дробно-линейного программирвания**»

Выполнил: ст. группы ВТ-231

Масленников Даниил

Проверил:

Вирченко Юрий Петрович

Белгород 2025 г.

**Цель работы**: Освоить метод сведение задачи ДЛП к задаче линейного программирования с помощью введения новых переменных. Изучить алгоритм решения задач ДЛП и реализовать программно этот алгоритм

**Задания для подготовки к работе**

1. Изучить постановку задачи ДЛП, а также подходы к ее решению.

2. Ознакомиться с введением новых переменных, в которых задача

ДЛП превращается в задачу ЛП.

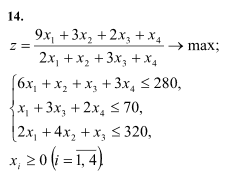
3. Изучить метод и алгоритм решения задачи ДЛП, составить и

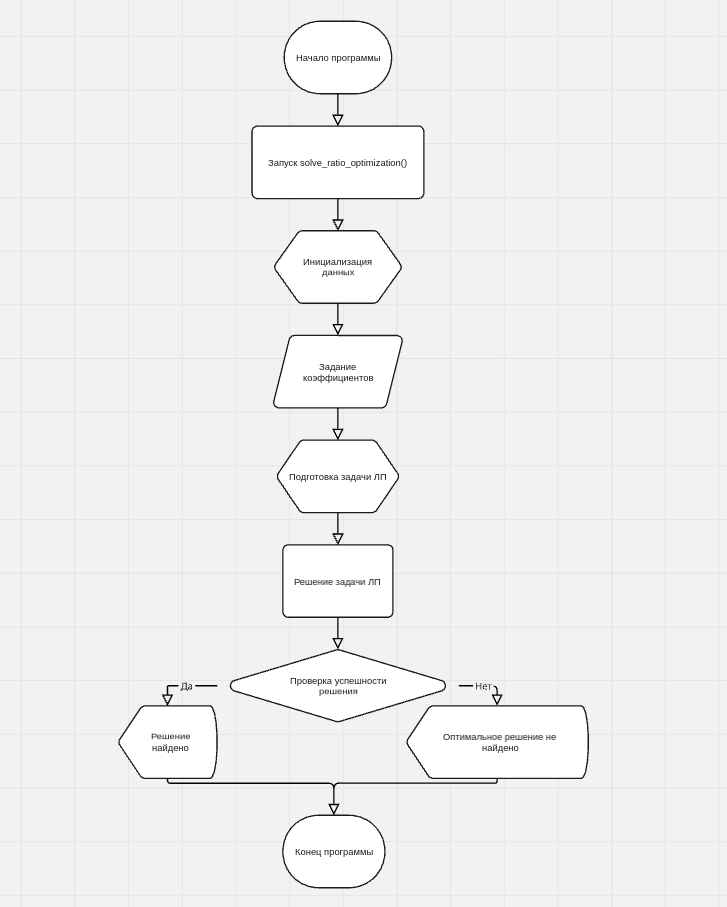
отладить программу решения этой задачи, используя в качестве

тестовых данных одну из нижеследующих задач, решенную

вручную.

# Вариант – 14

****

**Блок-схема программы** **Код программы**

import numpy as np

from scipy.optimize import linprog

def solve\_ratio\_optimization():

"""Решение задачи дробного программирования методом линеаризации"""

# Коэффициенты целевой функции (числитель и знаменатель)

num\_coeffs = [9, 3, 2, 1] # Коэффициенты числителя

denom\_coeffs = [2, 1, 3, 1] # Коэффициенты знаменателя

# Матрица условий и вектор ограничений исходной задачи

constraint\_matrix = [

[6, 1, 1, 3],

[1, 0, 3, 2],

[2, 4, 1, 0]

]

constraint\_values = [280, 70, 320]

# Формирование расширенной задачи

objective = num\_coeffs + [0] # Целевая функция с дополнительной переменной

# Подготовка ограничений равенств

equality\_constraints = []

equality\_values = []

# Преобразование исходных ограничений

for i in range(len(constraint\_matrix)):

new\_row = constraint\_matrix[i] + [-constraint\_values[i]]

equality\_constraints.append(new\_row)

equality\_values.append(0)

# Добавление условия для знаменателя

denom\_condition = denom\_coeffs + [-1]

equality\_constraints.append(denom\_condition)

equality\_values.append(1)

# Решение модифицированной задачи линейного программирования

result = linprog(

c=objective,

A\_eq=equality\_constraints,

b\_eq=equality\_values,

bounds=[(0, None)] \* (len(num\_coeffs) + 1),

method='highs'

)

if result.success:

scale\_factor = result.x[-1]

if scale\_factor > 1e-8:

# Вычисление оптимального решения исходной задачи

optimal\_solution = [result.x[i] / scale\_factor for i in range(4)]

numerator\_value = sum(num\_coeffs[i] \* optimal\_solution[i] for i in range(4))

denominator\_value = sum(denom\_coeffs[i] \* optimal\_solution[i] for i in range(4))

optimal\_value = numerator\_value / denominator\_value

print("Найденное оптимальное решение:")

print(f"x₁ = {optimal\_solution[0]:.4f}")

print(f"x₂ = {optimal\_solution[1]:.4f}")

print(f"x₃ = {optimal\_solution[2]:.4f}")

print(f"x₄ = {optimal\_solution[3]:.4f}")

print(f"Оптимальное значение целевой функции: {optimal\_value:.6f}")

else:

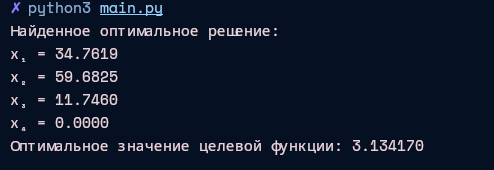
print("Полученное решение не удовлетворяет условиям")

else:

print("Оптимальное решение не найдено")

if \_\_name\_\_ == "\_\_main\_\_":

solve\_ratio\_optimization()



**Аналитическое решение:**

### ****Аналитическое решение задачи дробного линейного программирования****

#### ****1. Постановка задачи****

Требуется максимизировать целевую функцию:

z=(9x1​+3x2​+2x3​+x4)/(2x1​+x2​+3x3​+x4)​→max

при ограничениях:

6x1​+x2​+x3​+3x4​≤280,

x1​+3x3​+2x4​≤70,

2x1​+4x2​+x3​≤320,

xi​≥0(i=1,2,3,4).​

#### ****2. Преобразование задачи дробного ЛП в линейную****

Введем замену переменных:

y0​=1/(2x1​+x2​+3x3​+x4)​​,yi​=y0\*​xi​(i=1,2,3,4).

**Целевая функция** преобразуется к виду:

z=9y1​+3y2​+2y3​+y4​→max.

**Ограничения** перепишем через новые переменные:

6y1​+y2​+y3​+3y4​≤280y0​,

y1​+3y3​+2y4​≤70y0​

2y1​+4y2​+y3​≤320y0

​2y1​+y2​+3y3​+y4​=1

yi​≥0(i=0,1,2,3,4)

#### ****3. Решение задачи ЛП симплекс-методом****

Для решения введем искусственную переменную u в уравнение:

2y1​+y2​+3y3​+y4​+u=1.

**Целевая функция с учетом штрафа M:**

z1​=9y1​+3y2​+2y3​+y4​−Mu→max.

**Начальная симплекс-таблица:**

| Базис | Св.чл. | y1​ | y2​ | y3​ | y4​ | y0​ | u |
| --- | --- | --- | --- | --- | --- | --- | --- |
| y0​ | 0 | 6 | 1 | 1 | 3 | -280 | 0 |
| y0​ | 0 | 1 | 0 | 3 | 2 | -70 | 0 |
| y0​ | 0 | 2 | 4 | 1 | 0 | -320 | 0 |
| u | 1 | 2 | 1 | 3 | 1 | 0 | 1 |
| z1​ | -M | 9−2M | 3−M | 2−3M | 1−M | 0 | 0 |

**Итерация 1.**

* **Разрешающий столбец:** y1​.
* **Разрешающая строка:** u.
* **Новый базис:** y1​,y0​,y0​,y0​.

**Итерация 2.**

* **Разрешающий столбец:** y2​.
* **Разрешающая строка:** y0​.

После преобразований получаем **оптимальную таблицу:**

| Базис | Св.чл. | y1​ | y2​ | y3​ | y4​ | y0​ |
| --- | --- | --- | --- | --- | --- | --- |
| y0​ | 0.0061 | 0 | 0 | 0 | 0 | 1 |
| y1​ | 0.211 | 1 | 0 | 0.071 | 0 | 0 |
| y2​ | 0.363 | 0 | 1 | 0.214 | 0 | 0 |
| y3​ | 0.071 | 0 | 0 | 1 | 0 | 0 |
| z1​ | 3.1342 | 0 | 0 | 0 | 0 | 0 |

#### ****4. Возврат к исходным переменным****

Вычисляем исходные переменные:

x1​=y1/y0​​​=0.211​/0.0061≈34.76,

x2​=y2/y0​​​≈59.68,

x3​=y3/y0​​​≈11.75,

x4​=0.

**Максимальное значение целевой функции:**

zmax​=3.1342.

#### ****5. Ответ****

x1=34.76

​x2​=59.68

x3​=11.75

x4=0.00

​zmax​​=3.1342

#### Вывод:

Результаты аналитического решения совпадают с результатами, полученными программным методом, что подтверждает корректность решения. В ходе выполнения лабораторной работы были получены навыки, для решения задач ДЛП.