

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИТМО»**

Факультет информационных технологий и программирования

Дисциплина:

«Прикладная математика»

Отчёт по лабораторной работе №1

«Линейное программирование»

Группа М33001

Выполнили:

Соловьев Роман Сергеевич

Халилов Роман Эдуардович

Преподаватель:

Москаленко Мария

Александровна

Санкт-Петербург

2023 г.

Цель работы:

Изучение симплекс-метода решения задач линейного программирования.

Постановка задачи:

Необходимо реализовать парсер общей формы задачи линейного программирования. Разработать метод приведения к канонической форме. Реализовать симплекс-метод решения задачи.

Ход работы:

Весь код, использованный в ходе выполнения работы: [GitHub](#)

Часть 1 – парсинг:

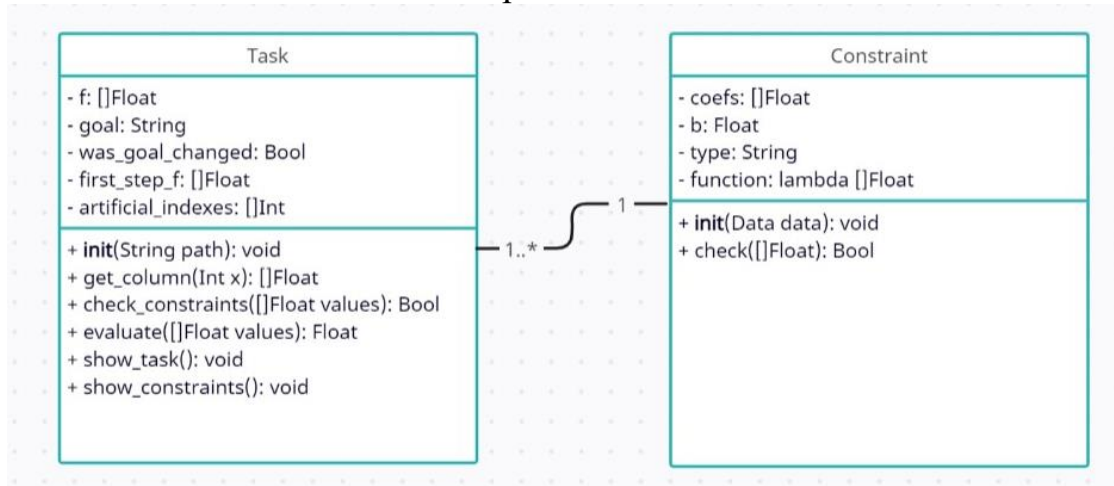
Принимаем файл формата json:

```
1  {
2    "f": [5000 , 2500] ,
3
4    "goal": "max" ,
5
6    "constraints": [
7      {
8        "coefs": [4 , 1.5] ,
9        "type": "leq" ,
10       "b": 24} ,
11
12      {
13        "coefs": [1200 , 150] ,
14        "type": "leq" ,
15        "b": 6000} ,
16
17      {
18        "coefs": [20 , 20] ,
19        "type": "leq" ,
20        "b": 200},
21
22      {
23        "coefs": [1 , 0] ,
24        "type": "geq" ,
25        "b": 2}
26    ]
27  }
```

Красивый (относительно) вывод спаршенного json'а:

```
Task:
5000 * X_1 + 2500 * X_2 → max
Constraints:
4 * X_1 + 1.5 * X_2 ≤ 24
1200 * X_1 + 150 * X_2 ≤ 6000
20 * X_1 + 20 * X_2 ≤ 200
1 * X_1 ≥ 2
```

Имеем два класса – задача и ограничение.



Часть 2 – переход к канонической форме:

Приводим задачу к канонической форме:

```
Task:
5000.0 * X_1 + 2500.0 * X_2 → max
Constraints:
4.0 * X_1 + 1.5 * X_2 + 1.0 * X_3 = 24
1200.0 * X_1 + 150.0 * X_2 + 1.0 * X_4 = 6000
20.0 * X_1 + 20.0 * X_2 + 1.0 * X_5 = 200
1.0 * X_1 - 1.0 * X_6 + 1.0 * X_7 = 2
```

- Превратили все неравенства в уравнения, добавив вспомогательные переменные.
- Если задание было найти минимум – модернизировали в задачу нахождения максимума с коэффициентами, домноженными на -1.
- Если в каком-либо ограничении свободный коэффициент был отрицательным – домножили всё уравнение на -1.
- Если для какого-то ограничения не было переменной, имеющей коэффициент, равный 1, и не входящей во все остальные ограничения, то добавляем искусственные переменные, удовлетворяющие этому условию.

Часть 3 – симплекс-метод (двухфазный):

Составляем симплекс-таблицу, используя вместо целевой функции вспомогательную – $f' = \sum x_i$, где i – индексы искусственных переменных.

Используем симплекс-метод с данной функцией. Таким образом мы избавляемся от искусственных переменных в базисе и переходим ко второй фазе. В ней мы используем целевую функцию и получаем ответ.

Симплекс-таблица до первой фазы:

Basis	b	1	2	3	4	5	6	7
3	24.0	4.0	1.5	1.0	0.0	0.0	0.0	0.0
4	6000.0	1200.0	150.0	0.0	1.0	0.0	0.0	0.0
5	200.0	20.0	20.0	0.0	0.0	1.0	0.0	0.0
7	2.0	1.0	0.0	0.0	0.0	0.0	-1.0	1.0
C	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
0.0	f	5000.0	2500.0	0.0	0.0	0.0	0.0	0.0
2.0	f'	0	0	0	0	0	0	1

Симплекс-таблица между фазами:

Basis	b	1	2	3	4	5	6	7
3	16.0	0.0	1.5	1.0	0.0	0.0	4.0	-4.0
4	3600.0	0.0	150.0	0.0	1.0	0.0	1200.0	-1200.0
5	160.0	0.0	20.0	0.0	0.0	1.0	20.0	-20.0
1	2.0	1.0	0.0	0.0	0.0	0.0	-1.0	1.0
C	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
10000.0	f	5000.0	2500.0	0.0	0.0	0.0	0.0	0.0
0	f'	0	0	0	0	0	0	1

Симплекс-таблица после второй фазы:

Basis	b	1	2	3	4	5	6	7
2	6.4	0.0	1.0	-0.3999999999999999	0.0	0.0799999999999999	0.0	0.0
6	1.6	0.0	0.0	0.3999999999999999	0.0	-0.02999999999999995	1.0	-1.0
4	719.9999999999999	0.0	0.0	-419.99999999999994	1.0	23.999999999999996	0.0	0.0
1	3.6	1.0	0.0	0.3999999999999999	0.0	-0.02999999999999995	0.0	0.0
C	34000.0	0.0	0.0	-1000.0000000000002	0.0	-49.999999999999986	0.0	0.0
34000.0	f	5000.0	2500.0	0.0	0.0	0.0	0.0	0.0
0	f'	0	0	0	0	0	0	1

Часть 4 – тестирование:

Протестировали работоспособность кода на различных случаях, в том числе при бесконечном множестве решений, при решении, включающем в себя переменные, стремящиеся в бесконечность и при отсутствии каких-либо решений.

Общий вывод по проделанной работе:

В ходе работы были реализованы чтение из файла формата json, приведение общей формы задачи линейного программирования к канонической и двухфазный симплекс-метод её решения.