

Министерство науки и образования РФ
Федеральное государственное бюджетное учреждение
высшего образования
«Тверской государственный технический университет»
(ТвГТУ)
Кафедра программного обеспечения

Отчет по лабораторной работе №1

По дисциплине: «Компьютерные системы моделирования»
Тема: «Построение аналитической модели по текстовому описанию
задачи и по результатам экспериментов»

Выполнил:
студент группы
Б.ПИН.РИС - 23.06
Протасов А.А.

Проверила:
старший преподаватель
кафедры ПО
Лисничук А.Б.

Тверь 2026

Введение

Цель работы: научиться строить математическую модель по постановке задачи и по табличным данным с помощью Python и Excel.

Математическая модель — это формальное описание реального объекта, процесса или явления на языке математики. Она включает:

Переменные — величины, которые могут изменяться.

Ограничения — уравнения или неравенства, описывающие связи между переменными.

Целевую функцию — показатель, который нужно оптимизировать (максимизировать или минимизировать).

Математические модели позволяют анализировать сложные системы, прогнозировать их поведение и находить оптимальные решения. В данной работе строится модель линейного программирования, где все зависимости линейны.

Построение аналитической модели по текстовому описанию данных

Постановка задачи

Для поддержания нормальной жизнедеятельности человеку ежедневно необходимо потреблять не менее:

- 118 г белков
- 56 г жиров
- 500 г углеводов
- 8 г минеральных солей

Известно содержание питательных веществ в 1 кг каждого из 7 продуктов и их цена. Требуется составить дневной рацион минимальной общей стоимости, удовлетворяющий всем нормам потребления.

Построение модели

Переменные:

- x_1 - количество мяса (кг)
- x_2 - количество рыбы (кг)
- x_3 - количество молока (кг)
- x_4 - количество масла (кг)
- x_5 - количество сыра (кг)
- x_6 - количество крупы (кг)
- x_7 - количество картофеля (кг)

Целевая функция (минимизация стоимости):

$$F(x) = 450x_1 + 350x_2 + 90x_3 + 180x_4 + 600x_5 + 80x_6 + 40x_7 \rightarrow \min$$

Ограничения:

$$180x_1 + 190x_2 + 30x_3 + 10x_4 + 260x_5 + 130x_6 + 21x_7 \geq 118 \text{ (белки)}$$

$$20x_1 + 3x_2 + 40x_3 + 865x_4 + 310x_5 + 30x_6 + 2x_7 \geq 56 \text{ (жиры)}$$

$$0x_1 + 0x_2 + 50x_3 + 6x_4 + 20x_5 + 650x_6 + 200x_7 \geq 500 \text{ (углеводы)}$$

$$9x_1 + 10x_2 + 7x_3 + 12x_4 + 60x_5 + 20x_6 + 10x_7 \geq 8 \text{ (минеральные соли)}$$

Реализация

| Продукт | Мясо | Рыба | Молоко | Масло | Сыр | Крупа | Картофель | Итого | Ед.изм | Норма |
|------------------|------|------|--------|-------|-----|--------|-----------|--------|--------|-------|
| Количество (кг) | 0 | 0 | 0 | 0,034 | 0 | 0,9051 | 0 | | | |
| Белки | 180 | 190 | 30 | 10 | 260 | 130 | 21 | 118 | г | 118 |
| Жиры | 20 | 3 | 40 | 865 | 310 | 30 | 2 | 56,563 | г | 56 |
| Углеводы | 0 | 0 | 50 | 6 | 20 | 650 | 200 | 588,52 | г | 500 |
| Минеральные соли | 9 | 10 | 7 | 12 | 60 | 20 | 10 | 18,51 | г | 8 |
| Цена 1 кг | 450 | 350 | 90 | 180 | 600 | 80 | 40 | | руб | |
| Общая стоимость | | | | | | | | 78,528 | руб | |
| Проверка норм | | | | | | | | | | |
| Белки | Да | | | | | | | | | |
| Жиры | Да | | | | | | | | | |
| Углеводы | Да | | | | | | | | | |
| Соли | Да | | | | | | | | | |

Рисунок 1 - Реализация в Excel, с помощью Auto Solver.



Рисунок 2 - Диаграмма потребляемых питательных веществ.

Решение на Python с помощью библиотеки pulp

```
from pulp import LpVariable, LpProblem, LpMinimize, value import time

def main(): print("ЗАДАЧА 1: ОПТИМИЗАЦИЯ РАЦИОНА ПИТАНИЯ
(реалистичные цены) ")

x1          =          LpVariable("Мясо_кг",          lowBound=0)
x2          =          LpVariable("Рыба_кг",          lowBound=0)
x3          =          LpVariable("Молоко_кг",        lowBound=0)
x4          =          LpVariable("Масло_кг",          lowBound=0)
x5          =          LpVariable("Сыр_кг",            lowBound=0)
x6          =          LpVariable("Крупа_кг",          lowBound=0)
x7          =          LpVariable("Картофель_кг",      lowBound=0)

#                      Создание                      задачи
problem      =          LpProblem('Минимальная_стоимость_рациона',
LpMinimize)

problem += 450*x1 + 350*x2 + 90*x3 + 180*x4 + 600*x5 + 80*x6 +
40*x7,          "Стоимость"

#                      Ограничения
problem += 180*x1 + 190*x2 + 30*x3 + 10*x4 + 260*x5 + 130*x6 +
21*x7          >=          118,          "Белки"
problem += 20*x1 + 3*x2 + 40*x3 + 865*x4 + 310*x5 + 30*x6 + 2*x7
>=          56,          "Жиры"
problem += 0*x1 + 0*x2 + 50*x3 + 6*x4 + 20*x5 + 650*x6 + 200*x7
>=          500,          "Углеводы"
problem += 9*x1 + 10*x2 + 7*x3 + 12*x4 + 60*x5 + 20*x6 + 10*x7
>=          8,          "Соли"
problem.solve()

#                      Результаты
```

```

print("\n" + " " * 70)
print("ОПТИМАЛЬНЫЙ РАЦИОН С РЕАЛИСТИЧНЫМИ ЦЕНАМИ")
print("=" * 70)

products = [
    ("Мясо", x1, 450, "руб/кг"),
    ("Рыба", x2, 350, "руб/кг"),
    ("Молоко", x3, 90, "руб/литр"),
    ("Масло", x4, 180, "руб/кг"),
    ("Сыр", x5, 600, "руб/кг"),
    ("Крупа", x6, 80, "руб/кг"),
    ("Картофель", x7, 40, "руб/кг")
]

print(f"\n{'ПРОДУКТ':<12} {'КОЛИЧЕСТВО':<15} {'ЦЕНА':<15}"
      {'СТОИМОСТЬ':<15}")
print("-" * 57)

total_cost = 0
daily_cost = 0
monthly_cost = 0

for name, var, price, unit in products:
    quantity = var.varValue
    if quantity > 0.0001:
        cost = quantity * price
        total_cost += cost
        print(f"{name:<12} {quantity:<15.4f} {price:<15}"
              {cost:<15.2f}")

print("-" * 57)
print(f"{'ВСЕГО':<12} {'':<15} {'':<15} {total_cost:<15.2f}")

# Расчет на месяц (30 дней)
daily_cost = total_cost
monthly_cost = daily_cost * 30

print(f"\nСтоимость питания в день: {daily_cost:.2f} руб.")
print(f"Стоимость питания в месяц (30 дней): {monthly_cost:.2f} руб.")

# Проверка норм
print("\n" + " " * 70)
print("ПРОВЕРКА ВЫПОЛНЕНИЯ СУТОЧНЫХ НОРМ")
print("=" * 70)

proteins = 180*x1.varValue + 190*x2.varValue + 30*x3.varValue +
10*x4.varValue + 260*x5.varValue + 130*x6.varValue +
21*x7.varValue

```

```

fats = 20*x1.varValue + 3*x2.varValue + 40*x3.varValue +
865*x4.varValue + 310*x5.varValue + 30*x6.varValue +
2*x7.varValue
carbs = 0*x1.varValue + 0*x2.varValue + 50*x3.varValue +
6*x4.varValue + 20*x5.varValue + 650*x6.varValue +
200*x7.varValue
salts = 9*x1.varValue + 10*x2.varValue + 7*x3.varValue +
12*x4.varValue + 60*x5.varValue + 20*x6.varValue +
10*x7.varValue

```

```

norms = [
    ("Белки", proteins, 118),
    ("Жиры", fats, 56),
    ("Углеводы", carbs, 500),
    ("Минеральные соли", salts, 8)
]

```

```

for name, obtained, required in norms:
    percent = (obtained / required) * 100
    status = "ДА" if obtained >= required / 1.000001 else "НЕТ"
    print(f"{name:<20} {obtained:>6.1f} г / {required:>3} г =
{percent:>6.1f}% {status}")

```

```

# Анализ стоимости питания
print("\n" + "==" * 70)
print("АНАЛИЗ СТОИМОСТИ ПИТАНИЯ")
print("=" * 70)

```

```

# Средняя стоимость 100 г белка
print(f"\nСтоимость 100 г белка из разных продуктов:")
print(f"• Картофель: {40/21*100:.1f} руб.")
print(f"• Крупа: {80/130*100:.1f} руб.")
print(f"• Молоко: {90/30*100:.1f} руб.")
print(f"• Рыба: {350/190*100:.1f} руб.")
print(f"• Мясо: {450/180*100:.1f} руб.")
print(f"• Сыр: {600/260*100:.1f} руб.")

```

```

if name == "main": main()

```