

1. Оптимизация производства.

Содержание задания:

Компания производит два продукта: А и В. Для их производства требуется два ресурса: трудовые часы и материалы. Каждая единица продукта А требует 3 часа труда и 2 единицы материала, а каждая единица продукта В требует 2 часа труда и 5 единиц материала. Компания располагает 120 трудовыми часами и 150 единицами материала. Прибыль от продажи каждой единицы продукта А составляет 40 у.е., а от продажи В — 60.

Какое количество продуктов А и В следует произвести, чтобы максимизировать прибыль? Формализуйте задачу как задачу линейного программирования и решите ее.

Пусть:

- x_1 - количество продукта А, которое было произведено компанией,
- x_2 - количество продукта В, которое было произведено компанией.

Тогда, задачу можно записать следующим образом, как задачу линейного программирования:

$$40x_1 + 60x_2 \rightarrow \max$$

$$3x_1 + 2x_2 \leq 120$$

$$2x_1 + 5x_2 \leq 150$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

x_1 и x_2 должны быть целыми числами

Код на языке python с использованием библиотеки cvxpy:

```
import cvxpy as cp

if __name__ == "__main__":
    x1 = cp.Variable(integer=True, nonneg=True)
    x2 = cp.Variable(integer=True, nonneg=True)
    objective = cp.Maximize(40 * x1 + 60 * x2)
    constraints = [
        3 * x1 + 2 * x2 <= 120,
        2 * x1 + 5 * x2 <= 150
    ]
    problem = cp.Problem(objective, constraints)
    problem.solve()
    print(f"Status: {problem.status}")
    print(f"Optimal value (max profit): {problem.value:.2f}")
    print(f"x1 (Product 1): {x1.value:.2f}")
    print(f"x2 (Product 2): {x2.value:.2f}")
    print("Check constraint:")
    print(f"3 * x1 + 2 * x2 = {(3 * x1 + 2 * x2).value} <= 120")
    print(f"2 * x1 + 5 * x2 = {(2 * x1 + 5 * x2).value} <= 150")
```

Запускаю этот код на выполнение - получаю следующий результат:

```
Status: optimal
Optimal value (max profit): 2220.00
x1 (Product 1): 27.00
x2 (Product 2): 19.00
Check constraint:
3 * x1 + 2 * x2 = 119.0 <= 120
2 * x1 + 5 * x2 = 149.0 <= 150
```

Итог:

Компания производит 27 единиц продукта А и 19 единиц продукта В и получает прибыль в 2220 у.е.

2. Проблема диеты

Содержание задания:

Человек должен потреблять три ключевых питательных вещества: белки, жиры и углеводы. Требования: не менее 50 г белков, 30 г жиров и 100 г углеводов в день. Доступны три вида продуктов с информацией о содержании веществ и стоимости:

Продукт	Белки (г)	Жиры (г)	Углеводы (г)	Цена за единицу
Продукт 1	5	2	10	3 у.е.
Продукт 2	10	5	20	5 у.е.
Продукт 3	15	7	25	8 у.е.

Какое количество каждого продукта следует включить в рацион, чтобы минимизировать стоимость при выполнении всех требований по питательным веществам? Формализуйте задачу как задачу линейного программирования и решите ее.

Пусть:

- x_1 - количество продукта 1 в рационе,
- x_2 - количество продукта 2 в рационе,
- x_3 - количество продукта 3 в рационе.

Тогда, задачу можно записать следующим образом, как задачу линейного программирования:

$$3x_1 + 5x_2 + 8x_3 \rightarrow \min$$

$$5x_1 + 10x_2 + 15x_3 \geq 50$$

$$2x_1 + 5x_2 + 7x_3 \geq 30$$

$$10x_1 + 20x_2 + 25x_3 \geq 100$$

$$x_1 \geq 0$$

$$x_2 \geq 0$$

$$x_3 \geq 0$$

x_1 , x_2 и x_3 должны быть целыми числами

Код на языке python с использованием библиотеки cvxpy:

```
import cvxpy as cp

if __name__ == "__main__":
    x1 = cp.Variable(integer=True, nonneg=True)
    x2 = cp.Variable(integer=True, nonneg=True)
    x3 = cp.Variable(integer=True, nonneg=True)
    objective = cp.Minimize(3 * x1 + 5 * x2 + 8 * x3)
    constraints = [
        5 * x1 + 10 * x2 + 15 * x3 >= 50,
        2 * x1 + 5 * x2 + 7 * x3 >= 30,
        10 * x1 + 20 * x2 + 25 * x3 >= 100
    ]
    problem = cp.Problem(objective, constraints)
    problem.solve()
    print(f"Status: {problem.status}")
    print(f"Optimal value (min cost): {problem.value:.2f}")
    print(f"x1 (Product 1): {x1.value:.2f}")
    print(f"x2 (Product 2): {x2.value:.2f}")
    print(f"x2 (Product 3): {x3.value:.2f}")
    print("Check constraint:")
    print(f"5 * x1 + 10 * x2 + 15 * x3 = {(5 * x1 + 10 * x2 + 15 * x3).value} >= 50")
    print(f"2 * x1 + 5 * x2 + 7 * x3 = {(2 * x1 + 5 * x2 + 7 * x3).value} >= 30")
    print(f"10 * x1 + 20 * x2 + 25 * x3 = {(10 * x1 + 20 * x2 + 25 * x3).value} >= 100")
```

Запускаю этот код на выполнение - получаю следующий результат:

```
Status: optimal
Optimal value (min cost): 30.00
x1 (Product 1): -0.00
x2 (Product 2): 6.00
x2 (Product 3): -0.00
Check constraint:
5 * x1 + 10 * x2 + 15 * x3 = 60.0 >= 50
2 * x1 + 5 * x2 + 7 * x3 = 30.0 >= 30
10 * x1 + 20 * x2 + 25 * x3 = 120.0 >= 100
```

Итог:

Достаточно потреблять 6 единиц продукта 2 в день. Стоить это будет 30 у.е.

3. Транспортная задача

Содержание задания:

Компания производит товары на трех фабриках и поставляет их в пять магазинов. Количество доступного товара на каждой фабрике: 100, 150 и 200 единиц. Требования магазинов: 80, 90, 110, 120 и 50 единиц соответственно. Стоимость транспортировки одной единицы товара с каждой фабрики в каждый магазин указана в условных единицах в таблице ниже:

	Магазин 1	Магазин 2	Магазин 3	Магазин 4	Магазин 5
Фабрика 1	4	3	6	7	9
Фабрика 2	5	4	8	6	7
Фабрика 3	9	7	4	5	6

Как распределить поставки, чтобы минимизировать затраты на транспортировку, удовлетворяя все требования? Формализуйте задачу как задачу линейного программирования и решите ее.

Пусть:

- x_{11} - количество товара, отправленное с фабрики 1 в магазин 1,
- x_{12} - количество товара, отправленное с фабрики 1 в магазин 2,
- x_{13} - количество товара, отправленное с фабрики 1 в магазин 3,
- x_{14} - количество товара, отправленное с фабрики 1 в магазин 4,
- x_{15} - количество товара, отправленное с фабрики 1 в магазин 5,
- x_{21} - количество товара, отправленное с фабрики 2 в магазин 1,
- x_{22} - количество товара, отправленное с фабрики 2 в магазин 2,
- x_{23} - количество товара, отправленное с фабрики 2 в магазин 3,
- x_{24} - количество товара, отправленное с фабрики 2 в магазин 4,
- x_{25} - количество товара, отправленное с фабрики 2 в магазин 5,
- x_{31} - количество товара, отправленное с фабрики 3 в магазин 1,
- x_{32} - количество товара, отправленное с фабрики 3 в магазин 2,
- x_{33} - количество товара, отправленное с фабрики 3 в магазин 3,
- x_{34} - количество товара, отправленное с фабрики 3 в магазин 4,
- x_{35} - количество товара, отправленное с фабрики 3 в магазин 5.

Тогда, задачу можно записать следующим образом, как задачу линейного программирования:

$$4x_{11} + 3x_{12} + 6x_{13} + 7x_{14} + 9x_{15} + 5x_{21} + 4x_{22} + 8x_{23} + 6x_{24} + 7x_{25} + 9x_{31} + 7x_{32} + 4x_{33} + 5x_{34} + 6x_{35} \rightarrow \min$$

$$x_{11} + x_{12} + x_{13} + x_{14} + x_{15} \leq 100$$

$$x_{21} + x_{22} + x_{23} + x_{24} + x_{25} \leq 150$$

$$x_{31} + x_{32} + x_{33} + x_{34} + x_{35} \leq 200$$

$$x_{11} + x_{21} + x_{31} = 80$$

$$x_{12} + x_{22} + x_{32} = 90$$

$$x_{13} + x_{23} + x_{33} = 110$$

$$x_{14} + x_{24} + x_{34} = 120$$

$$x_{15} + x_{25} + x_{35} = 50$$

$$x_{11} \geq 0$$

$$x_{12} \geq 0$$

$$x_{13} \geq 0$$

$$x_{14} \geq 0$$

$$x_{15} \geq 0$$

$$x_{21} \geq 0$$

$$x_{22} \geq 0$$

$$x_{23} \geq 0$$

$$x_{24} \geq 0$$

$$x_{25} \geq 0$$

$$x_{31} \geq 0$$

$$x_{32} \geq 0$$

$$x_{33} \geq 0$$

$$x_{34} \geq 0$$

$$x_{35} \geq 0$$

Код на языке python с использованием библиотеки cvxpy:

```
import cvxpy as cp

if __name__ == "__main__":
    x11 = cp.Variable(integer=True, nonneg=True)
    x12 = cp.Variable(integer=True, nonneg=True)
    x13 = cp.Variable(integer=True, nonneg=True)
    x14 = cp.Variable(integer=True, nonneg=True)
    x15 = cp.Variable(integer=True, nonneg=True)
    x21 = cp.Variable(integer=True, nonneg=True)
    x22 = cp.Variable(integer=True, nonneg=True)
    x23 = cp.Variable(integer=True, nonneg=True)
    x24 = cp.Variable(integer=True, nonneg=True)
    x25 = cp.Variable(integer=True, nonneg=True)
    x31 = cp.Variable(integer=True, nonneg=True)
    x32 = cp.Variable(integer=True, nonneg=True)
    x33 = cp.Variable(integer=True, nonneg=True)
    x34 = cp.Variable(integer=True, nonneg=True)
    x35 = cp.Variable(integer=True, nonneg=True)
    objective = cp.Minimize(
        4 * x11 + 3 * x12 + 6 * x13 + 7 * x14 + 9 * x15 +
```

```

5 * x21 + 4 * x22 + 8 * x23 + 6 * x24 + 7 * x25 +
9 * x31 + 7 * x32 + 4 * x33 + 5 * x34 + 6 * x35)
constraints = [
    x11 + x12 + x13 + x14 + x15 <= 100,
    x21 + x22 + x23 + x24 + x25 <= 150,
    x31 + x32 + x33 + x34 + x35 <= 200,
    x11 + x21 + x31 == 80,
    x12 + x22 + x32 == 90,
    x13 + x23 + x33 == 110,
    x14 + x24 + x34 == 120,
    x15 + x25 + x35 == 50
]
problem = cp.Problem(objective, constraints)
problem.solve()
print(f"Status: {problem.status}")
print(f"Optimal value (min cost): {problem.value:.2f}")
print(f"x11 (Product 1): {x11.value:.2f}")
print(f"x12 (Product 1): {x12.value:.2f}")
print(f"x13 (Product 1): {x13.value:.2f}")
print(f"x14 (Product 1): {x14.value:.2f}")
print(f"x15 (Product 1): {x15.value:.2f}")
print(f"x21 (Product 1): {x21.value:.2f}")
print(f"x22 (Product 1): {x22.value:.2f}")
print(f"x23 (Product 1): {x23.value:.2f}")
print(f"x24 (Product 1): {x24.value:.2f}")
print(f"x25 (Product 1): {x25.value:.2f}")
print(f"x31 (Product 1): {x31.value:.2f}")
print(f"x32 (Product 1): {x32.value:.2f}")
print(f"x33 (Product 1): {x33.value:.2f}")
print(f"x34 (Product 1): {x34.value:.2f}")
print(f"x35 (Product 1): {x35.value:.2f}")
print("Check constraint:")
print(f"x11 + x12 + x13 + x14 + x15 = {(x11 + x12 + x13 + x14 +
x15).value} <= 100")
print(f"x21 + x22 + x23 + x24 + x25 = {(x21 + x22 + x23 + x24 +
x25).value} <= 150")
print(f"x31 + x32 + x33 + x34 + x35 = {(x31 + x32 + x33 + x34 +
x35).value} <= 200")
print(f"x11 + x21 + x31 = {(x11 + x21 + x31).value} == 80")
print(f"x12 + x22 + x32 = {(x12 + x22 + x32).value} == 90")
print(f"x13 + x23 + x33 = {(x13 + x23 + x33).value} == 110")
print(f"x14 + x24 + x34 = {(x14 + x24 + x34).value} == 120")
print(f"x15 + x25 + x35 = {(x15 + x25 + x35).value} == 50")

```

Запускаю этот код на выполнение - получаю следующий результат:

```
Status: optimal
```

```
Optimal value (min cost): 2080.00
x11 (Product 1): 80.00
x12 (Product 1): 20.00
x13 (Product 1): -0.00
x14 (Product 1): -0.00
x15 (Product 1): -0.00
x21 (Product 1): -0.00
x22 (Product 1): 70.00
x23 (Product 1): -0.00
x24 (Product 1): 30.00
x25 (Product 1): 50.00
x31 (Product 1): -0.00
x32 (Product 1): -0.00
x33 (Product 1): 110.00
x34 (Product 1): 90.00
x35 (Product 1): -0.00
Check constraint:
x11 + x12 + x13 + x14 + x15 = 100.0 <= 100
x21 + x22 + x23 + x24 + x25 = 150.0 <= 150
x31 + x32 + x33 + x34 + x35 = 200.0 <= 200
x11 + x21 + x31 = 80.0 == 80
x12 + x22 + x32 = 90.0 == 90
x13 + x23 + x33 = 110.0 == 110
x14 + x24 + x34 = 120.0 == 120
x15 + x25 + x35 = 50.0 == 50
```

Итого:

С фабрики 1:

- в магазин 1 везем 80 единиц товара,
- в магазин 2 везем 20 единиц товара,
- в магазин 3 везем 0 единиц товара,
- в магазин 4 везем 0 единиц товара,
- в магазин 5 везем 0 единиц товара.\

С фабрики 2:

- в магазин 1 везем 0 единиц товара,
- в магазин 2 везем 70 единиц товара,
- в магазин 3 везем 0 единиц товара,
- в магазин 4 везем 30 единиц товара,
- в магазин 5 везем 50 единиц товара.

С фабрики 3:

- в магазин 1 везем 0 единиц товара,
- в магазин 2 везем 0 единиц товара,
- в магазин 3 везем 110 единиц товара,
- в магазин 4 везем 90 единиц товара,
- в магазин 5 везем 0 единиц товара.

Общие транспортные расходы - 2080 у.е.