

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Лабораторная работа №4
по дисциплине «Исследование операций»
Вариант 2

Бобовоза Владислава
Сергеевича
студента 3 курса, 6 группы
специальность «прикладная
математика»

Минск, 2024

Задача о рюкзаке (англ. Knapsack problem) — дано n предметов, предмет i имеет массу $w_i > 0$ и стоимость $p_i > 0$. Необходимо выбрать из этих предметов такой набор, чтобы суммарная масса не превосходила заданной величины W (вместимость рюкзака), а суммарная стоимость была максимальна.

Рассмотрим задачу Неограниченный рюкзак (англ. Unbounded Knapsack Problem), в которой любой предмет может быть выбран любое количество раз.

Формулировка Задачи

Каждый предмет может быть выбран любое число раз. Задача выбрать количество x_i предметов каждого типа так, чтобы:

- максимизировать общую стоимость: $\sum_{i=1}^n p_i x_i$;
- выполнялось условие совместности: $\sum_{i=1}^n w_i x_i \leq W$;
- где $x_i \geq 0$ целое, для всех $i = 1, 2, \dots, n$.

Условие индивидуальной задачи:

$$\begin{aligned} \max & 3x_1 + 8x_2 + 12x_3 \\ \text{s. t.} & 2x_1 + 3x_2 + 3x_3 \leq 8 \\ & x_1, x_2, x_3 \geq 0 - \text{целые} \end{aligned}$$

Листинг программы на Python:

```
import numpy as np

def knapsack(v, w, W):
    size = len(v)
    f = np.zeros((size + 1, W + 1))
    p = np.zeros((size + 1, W + 1))

    for i in range(1, size + 1):
        for j in range(1, W + 1):
            if w[i - 1] <= j:
                f[i,j] = max(f[i - 1, j], f[i, j - w[i - 1]] + v[i - 1])
            else:
                f[i,j] = f[i - 1, j]

            if f[i,j] == f[i - 1, j]:
                p[i,j] = 0
            else:
                p[i,j] = 1

    # Обратный ход
    selected = np.zeros_like(w)
```

```

i, j = size, W
while i != 0 and j != 0:
    if p[i,j] == 1:
        selected[i - 1] += 1
        j -= w[i - 1]
    else:
        i -= 1

return f[size, W], selected

if __name__ == '__main__':
    v = np.array([3, 8, 12])
    w = np.array([2, 3, 3])
    W = 8

    max_value, selected = knapsack(v, w, W)
    print(f"Оптимальное решение задачи: {max_value}")
    print(f"Выбранные предметы      : x1={selected[0]}, x2={selected[1]},
x3={selected[2]}")

```

Результаты выполнения программы на Python:

```

Оптимальное решение задачи: 27.0
Выбранные предметы      : x1=1, x2=0, x3=2

```