

# Комбинаторная оптимизация

## Задача о рюкзаке

2 марта 2025 г.

### 1 Формулировка

Дано  $N$  предметов и рюкзак вместимости  $W$ . Каждый предмет описывается парой  $(c_i, w_i)$  — стоимостью и весом.

Набор  $I \subseteq \{1, \dots, N\}$  предметов называется *допустимым*, если выполнено условие

$$\sum_{i \in I} w_i \leq W$$

*Стоимостью* набора называется величина

$$C(I) = \sum_{i \in I} c_i$$

Допустимый набор называется *оптимальным*, если не существует допустимого набора с большей стоимостью.

Ваша задача заключается в поиске допустимого решения, максимального близкого по стоимости к оптимальному.

### 2 Входные данные

Первая строка входных данных содержит два целых числа  $N$  и  $W$  — количество предметов и максимальный вес рюкзака. В следующих  $N$  строках содержатся пары целых чисел  $(c_i, w_i)$ , описывающие предметы. Для всех тестов верно, что  $N \leq 200\,000$  и  $W \leq 2\,500\,000$ . Гарантируется, что стоимость любого допустимого решения помещается в стандартный тип `int`.

### 3 Формат вывода

Первая строка выходных данных должна содержать число  $C$  — стоимость вашего допустимого набора. Во второй строке должны быть перечислены номера предметов составляющих ваш набор, то есть набор различных чисел из диапазона  $[1, 2, \dots, n]$ .

## 4 Ограничения

- Ограничение по времени: 10с
- Ограничение по памяти: 512МВ

В качестве языка программирования разрешается использовать только C++ с версиями компиляторов из Яндекс Контеста. Использование сторонних библиотек для оптимизации запрещено — код оптимизационного движка должен быть написан вами.

## 5 Оценивание

Оценка за задачу состоит из оценки за качество вашего решения (не более 5 баллов) и оценки за отчет (не более 1 балла). При этом в случае, если стоимость вашего решения не превосходит стоимости жадного решения, получить больше 1 балла за задачу нельзя.

Оценивание качества вашего решения будет производиться на основе разности ответов вашего решения и оптимального решения. Суммарный балл будет получаться как сумма баллов по каждому из тестов. На заданном тесте оптимальное решение будет оцениваться в 0.5 балла, а жадное решение в 0.05 балла.

## 6 Полезные советы

- Для поиска оптимума на открытых тестах вы можете применить динамическое программирование (например, динамику по весу), оно будет долго работать, но вы узнаете оптимальное решение и будете понимать к чему надо стремиться.
- Начните с простого решения и старайтесь постепенно его усложнять. Попробуйте разные подходы, рассказанные на семинаре.
- В Яндекс Контесте баллы отождражаются как дробь  $\frac{your\_score}{greedy\_score}$ .

## 7 Технические подробности

В архиве предоставлен публичный набор тестов, на котором можно отлаживать свою программу и сравнивать свои результаты с другими участниками. Решение будет оцениваться на приватном наборе тестов, который по структуре похож на публичный. Тесты с 1 по 10 в последней задаче конкурса соответствуют публичным тестам, с 11 по 20 — приватным тестам. Мы отображаем время работы, чтобы вы понимали, на каждой из посылок в этой задаче, успело ли отработать решение.

В качестве решения вы должны прислать **ссылку на вашу посылку в Яндекс Контесте и отчет с описанием решения**. Описание идей и экспериментов (даже неудачных) поощряются.

## Содержимое архива

- `data/*` — набор публичных тестов
- `example.py` — пример решения, демонстрирующий формат ввода/вывода