

## Задача А. 0-1 рюкзак: наибольший вес

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дано  $n$  золотых слитков массой  $m_1, \dots, m_n$ . Ими наполняют рюкзак, который выдерживает вес не более  $w$ . Какую наибольшую массу золота можно унести в таком рюкзаке?

### Формат входных данных

В первой строке вводится натуральное число  $n$ , не превышающее 100, и натуральное число  $w$ , не превышающее 10000.

Во второй строке вводятся  $n$  натуральных чисел  $m_i$ , не превышающих 100.

### Формат выходных данных

Выведите одно целое число — наибольшую возможную массу золота, которую можно унести в данном рюкзаке.

### Примеры

| стандартный ввод      | стандартный вывод |
|-----------------------|-------------------|
| 2 3195<br>38 41       | 79                |
| 5 100<br>2 71 8 28 18 | 99                |

## Задача В. Гирьки

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дан набор гирек массой  $m_1, \dots, m_n$ . Можно ли их разложить на две чаши весов таким образом, чтобы они оказались в равновесии?

### Формат входных данных

Первая строка входных данных содержит натуральное число  $n$ , не превышающее 100. Далее идет  $n$  натуральных чисел  $m_i$ , не превышающих 100.

### Формат выходных данных

Программа должна вывести «yes», если гирьки можно разложить на две кучки равной массы, или «no» в противном случае.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 4<br>4 2 3 1     | YES               |

## Задача С. Сочетания-2

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

По данным натуральным  $k$  и  $n$  ( $1 \leq k \leq n$ ) выведите все **убывающие** последовательности длины  $k$  состоящие из чисел  $1 \dots n$  в лексикографическом порядке.

### Формат входных данных

Во входном файле два числа —  $k$  и  $n$  ( $1 \leq k \leq n \leq 1000$ ).

### Формат выходных данных

Каждая последовательность должна выводиться в отдельной строке, вывод должен завершаться символом новой строки. Числа, входящие в последовательность, должны быть разделены одним пробелом. Гарантируется, что количество чисел в выходном файле не превосходит 500 000.

### Пример

| стандартный ввод | стандартный вывод                                                                      |
|------------------|----------------------------------------------------------------------------------------|
| 3 5              | 3 2 1<br>4 2 1<br>4 3 1<br>4 3 2<br>5 2 1<br>5 3 1<br>5 3 2<br>5 4 1<br>5 4 2<br>5 4 3 |

## Задача D. Гирьки: три кучки

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Дан набор гирек массой  $m_1, \dots, m_N$ . Можно ли их разложить на три кучки равной массы?

### Формат входных данных

Первая строка входных данных содержит натуральное число  $N$ , не превышающее 60.

Во второй строке через пробел записаны  $N$  натуральных чисел  $m_i$ , не превышающих 60.

### Формат выходных данных

Выведите три строки, описывающие наборы гирек, либо число -1, если решение не существует.

В каждой строке сначала выведите количество гирек в соответствующем наборе, а затем через пробел номера гирек в наборе.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 5                | 2 4 1             |
| 4 2 3 1 5        | 2 3 2             |
|                  | 1 5               |

## Задача Е. Рюкзак максимальной стоимости

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Даны  $N$  предметов различной массы и стоимости. Ими наполняют рюкзак, который выдерживает вес не более  $S$ . Определите набор предметов, который можно унести в рюкзаке, имеющий наибольшую стоимость.

Асимптотика  $O(NS)$ .

### Формат входных данных

В первой строке через пробел вводятся натуральные числа  $N$  и  $S$ .

Во второй строке через пробел вводятся  $N$  натуральных чисел — массы предметов.

В третьей строке через пробел вводятся  $N$  натуральных чисел — стоимости предметов.

### Формат выходных данных

В первой строке выведите число  $k$  — количество предметов в рюкзаке, образующих наибольшую возможную стоимость. Во второй строке через пробел выведите  $k$  различных чисел от 1 до  $N$  — номера предметов.

### Пример

| стандартный ввод          | стандартный вывод |
|---------------------------|-------------------|
| 4 6<br>2 4 1 2<br>7 2 5 1 | 3<br>4 3 1        |

## Задача F. Коровы и монеты

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Коровы не только создали собственное правительство, но и решили создать свою собственную денежную систему. В своем бунтарском духе они любопытствуют по поводу значений монет. Традиционно монеты имеют значения как 1, 5, 10, 20 или 25, 50 и 100 единиц, иногда добавляется монета в 2 единицы для хорошей меры.

Коровы хотят узнать, сколько различных способов можно выдать определенную сумму денег, используя различные системы монет. Например, используя систему со значениями  $\{1, 2, 5, 10, \dots\}$ , можно создать 18 единиц несколькими разными способами, включая:  $18 \times 1$ ,  $9 \times 2$ ,  $8 \times 2 + 2 \times 1$ ,  $3 \times 5 + 2 + 1$  и многие другие.

Напишите программу, чтобы вычислить, сколько способов можно построить заданную сумму денег  $N$  ( $1 \leq N \leq 10,000$ ) с использованием  $V$  ( $1 \leq V \leq 25$ ) монет.

### Формат входных данных

В первой строке вводится два числа  $N$  и  $V$ .

В следующих  $V$  строках вводятся номиналы монет.

### Формат выходных данных

В единственной строке выведите ответ на задачу.

### Пример

| стандартный ввод    | стандартный вывод |
|---------------------|-------------------|
| 3 10<br>1<br>2<br>5 | 10                |

## Задача G. Большой рюкзак

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

У вас есть  $N$  предметов с весами  $w_1, w_2, \dots, w_N$ . Требуется проверить, можно ли выбрать некоторые предметы, суммарный вес которых равен  $W$ .

### Формат входных данных

В первой строке записаны два целых числа  $N$  и  $W$  ( $1 \leq N \leq 2\,500$ ,  $1 \leq W \leq 6\,250\,000$ ).

Во второй строке через пробел записаны  $N$  целых чисел  $w_1, w_2, \dots, w_N$  ( $1 \leq w_i \leq 2\,500$ ) — веса предметов.

### Формат выходных данных

В единственной строке выведите «YES» (без кавычек), если, используя данные предметы, можно набрать вес  $W$ , либо «NO» в противном случае.

### Примеры

| стандартный ввод  | стандартный вывод |
|-------------------|-------------------|
| 5 10<br>1 2 3 4 5 | YES               |
| 2 10<br>4 5       | NO                |

## Задача Н. Большой рюкзак, но теперь с восстановлением!

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

У вас есть  $N$  предметов с весами  $w_1, w_2, \dots, w_N$ . Требуется проверить, можно ли выбрать некоторые предметы, суммарный вес которых равен  $W$ .

### Формат входных данных

В первой строке записаны два целых числа  $N$  и  $W$  ( $1 \leq N \leq 2\,500$ ,  $1 \leq W \leq 6\,250\,000$ ).

Во второй строке через пробел записаны  $N$  целых чисел  $w_1, w_2, \dots, w_N$  ( $1 \leq w_i \leq 2\,500$ ) — веса предметов.

### Формат выходных данных

В первой строке выведите «YES» (без кавычек), если, используя данные предметы, можно набрать вес  $W$ , либо «NO» в противном случае.

В случае, если набрать вес  $W$  возможно, восстановите ответ.

Во второй строке выведите одно целое число  $K$  ( $1 \leq K \leq N$ ) — количество выбранных предметов.

В третьей строке выведите через пробел  $K$  различных целых чисел  $i_1, i_2, \dots, i_K$  ( $1 \leq i_j \leq N$ ) — номера выбранных предметов.

### Примеры

| стандартный ввод  | стандартный вывод   |
|-------------------|---------------------|
| 5 10<br>1 2 3 4 5 | YES<br>4<br>1 2 3 4 |
| 2 10<br>4 5       | NO                  |

## Задача I. Сокровища

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дочь короля Флатландии собирается выйти за прекрасного принца. Принц хочет подарить принцессе сокровища, но он не уверен какие именно бриллианты из своей коллекции выбрать.

В коллекции принца  $n$  бриллиантов, каждый характеризуется весом  $w_i$  и стоимостью  $v_i$ . Принц хочет подарить наиболее дорогие бриллианты, однако король умен и не примет бриллиантов суммарного веса больше  $R$ . С другой стороны, принц будет считать себя жадным всю оставшуюся жизнь, если подарит бриллиантов суммарным весом меньше  $L$ .

Помогите принцу выбрать набор бриллиантов наибольшей суммарной стоимости, чтобы суммарный вес был в отрезке  $[L, R]$ .

### Формат входных данных

Первая строка содержит число  $n$  ( $1 \leq n \leq 32$ ),  $L$  и  $R$  ( $0 \leq L \leq R \leq 10^{18}$ ). Следующие  $n$  строк описывают бриллианты и содержит по два числа — вес и стоимость соответствующего бриллианта ( $1 \leq w_i, v_i \leq 10^{15}$ ).

### Формат выходных данных

Первая строка вывода должна содержать  $k$  — количество бриллиантов, которые нужно подарить принцессе. Вторая строка должна содержать номера даримых бриллиантов.

Бриллианты нумеруются от 1 до  $n$  в порядке появления во входных данных.

Если составить подарок принцессе невозможно, то выведите 0 в первой строке вывода.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 3 6 8            | 1                 |
| 3 10             | 2                 |
| 7 3              |                   |
| 8 2              |                   |

## Задача J. Коровы

|                         |                   |
|-------------------------|-------------------|
| Имя входного файла:     | стандартный ввод  |
| Имя выходного файла:    | стандартный вывод |
| Ограничение по времени: | 1 секунда         |
| Ограничение по памяти:  | 256 мегабайт      |

Коровы строят американские горки! Они хотят вашей помощи, чтобы спроектировать как можно более веселые горки, при этом соблюдая бюджет.

Американские горки будут построены на длинной прямой полосе земли длиной  $L$  ( $1 \leq L \leq 1000$ ). У коров есть  $N$  различных вариантов компонент для их американских горок. Каждый компонент  $i$  имеет фиксированную длину  $W_i$  ( $1 \leq W_i \leq L$ ). Из-за различного рельефа каждый компонент  $i$  может быть построен только начиная от места  $X_i$  ( $0 \leq X_i \leq L - W_i$ ). Коровы хотят соединить различные компоненты горок, начиная с 0 и заканчивая на  $L$ , так что конец каждого компонента (кроме последнего) является началом следующего компонента.

Кроме того каждый компонент  $i$  имеет «рейтинг веселья»  $F_i$  ( $1 \leq F_i \leq 1,000,000$ ) и стоимость  $C_i$  ( $1 \leq C_i \leq 1000$ ). Общая веселость американских горок — это сумма веселья от каждого использованного компонента; общая стоимость также является суммой затрат каждого использованного компонента. Общий бюджет коров составляет  $B$  ( $1 \leq B \leq 1000$ ). Помогите коровам определить самые веселые американские горки, которые они могут построить в рамках своего бюджета.

### Формат входных данных

В первой строке вводятся три числа  $L, N, B$ .

В следующих  $N$  строках вводятся по 4 числа  $X_i, W_i, F_i$ , и  $C_i$ .

### Формат выходных данных

В единственной строке выведите одно число — максимальный суммарный рейтинг веселья американской горки, который может получить у коров. Если невозможно построить никакие горки в рамках бюджета, выведите  $-1$  в единственной строке.

### Пример

| стандартный ввод                                                           | стандартный вывод |
|----------------------------------------------------------------------------|-------------------|
| 5 6 10<br>0 2 20 6<br>2 3 5 6<br>0 1 2 1<br>1 1 1 3<br>1 2 5 4<br>3 2 10 2 | 17                |

## Задача К. Перестановка по номеру

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Выведите перестановку по её номеру.

### Формат входных данных

В первой строке входного файла записано число  $N$  ( $1 \leq N \leq 12$ ) — количество элементов в перестановке. Во второй строке записано число  $K$  ( $0 \leq K < N!$ ) — номер перестановки в нумерации с нуля.

### Формат выходных данных

В выходной файл выведите  $N$  чисел через пробел — искомую перестановку.

### Пример

| стандартный ввод | стандартный вывод |
|------------------|-------------------|
| 3<br>0           | 1 2 3             |

## Задача L. Номер правильной последовательности

Имя входного файла: `brackets2num2.in`  
Имя выходного файла: `brackets2num2.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Вам дана правильная скобочная последовательность с двумя типами скобок. Выведите её номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, '(', ')', '[', ]'. Последовательности занумерованы, начиная с 0. Количество открывающихся скобок в последовательности — от 1 до 20.

### Формат входных данных

Во входном файле записана строка из круглых и квадратных скобок. Длина строки не превосходит 20 символов.

### Формат выходных данных

Выведите одно число — ответ на задачу.

### Примеры

| <code>brackets2num2.in</code>  | <code>brackets2num2.out</code> |
|--------------------------------|--------------------------------|
| <code>()</code>                | 0                              |
| <code>[]</code>                | 1                              |
| <code>()()</code>              | 1                              |
| <code>()[]</code>              | 2                              |
| <code>([])()[]</code>          | 100                            |
| <code>[[[(())[]([]))]]]</code> | 266079                         |