

Задача о рюкзаке и её вариации.

Дан максимальный вес W и n пар чисел c_1, \dots, c_n и w_1, \dots, w_n — стоимости и веса n предметов.

Задача: выбрать набор предметов из указанных с суммарным весом не превосходящим W и максимальной общей стоимостью.

- предметы могут повторяться (“супермаркет”)

$A[w]$ — оптимальная стоимость предметов, которые можно поместить в рюкзак объёма w .

$$A[w] = \max_{w_i \leq w} \{A[w - w_i] + c_i\}, \text{ время работы } O(nW)$$

Сравнить с жадной стратегией: брать сначала предметы с максимальной удельной стоимостью.

- предметы не повторяются (“антикварная лавка” или 0-1 рюкзак)

$A[k, w]$ — оптимальная стоимость предметов, которые можно поместить в рюкзак объёма w при условии, что можно использовать только первые k предметов.

$$A[k, w] = \max\{A[k - 1, w], A[k - 1, w - w_k] + c_k\}$$

A. Разложение в сумму кубов

Дано натуральное число N . Необходимо представить его в виде суммы точных кубов, содержащей наименьшее число слагаемых. Программа должна вывести это число слагаемых. Программа получает на вход натуральное число N , не превосходящее 10^6 .

Программа должна вывести единственное натуральное число — ответ на вопрос задачи.

Input	Output
33	5

B. Рюкзак без повторов: точный вес

Дано N золотых слитков массой m_1, \dots, m_N . Ими наполняют рюкзак, который выдерживает вес не более M . Можно ли набрать вес в точности M ?

В первой строке вводится натуральное число N ($N < 100$) и натуральное число M ($M < 10^5$).

Во второй строке вводятся N натуральных чисел m_i ($m_i < 100$).

Выведите YES или NO.

Input	Output
1 5968 18	NO

C. Рюкзак без повторов: наибольший вес

Дано N золотых слитков массой m_1, \dots, m_N . Ими наполняют рюкзак, который выдерживает вес не более M . Какую наибольшую массу золота можно унести в таком рюкзаке?

В первой строке вводится натуральное число N ($N < 100$) и натуральное число M ($M < 10000$).

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

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

Input	Output
2 3195 38 41	79

D* Банкомат

В некотором государстве в обращении находятся банкноты определенных номиналов. Национальный банк хочет, чтобы банкомат выдавал любую запрошенную сумму при помощи минимального числа банкнот, считая, что запас банкнот каждого номинала неограничен. Помогите Национальному банку решить эту задачу.

Первая строка входных данных содержит натуральное число N не превосходящее 100 — количество номиналов банкнот в обращении. Вторая строка входных данных содержит N различных натуральных чисел x_1, x_2, \dots, x_N , не превосходящих 10^6 — номиналы банкнот.

Третья строчка содержит натуральное число S , не превосходящее 10^6 — сумму, которую необходимо выдать.

Программа должна найти представление числа S в виде суммы слагаемых из множества x_i , содержащее минимальное число слагаемых и вывести это представление на экран (в виде последовательности чисел, разделенных пробелами). Если таких представлений существует несколько, то программа должна вывести любое (одно) из них. Если такое представление не существует, то программа должна вывести одно число -1 .

Input	Output
5 1 3 7 12 32 40	32 7 1

Е. *0-1 рюкзак — максимальная стоимость*

Дано N предметов массой m_1, \dots, m_N и стоимостью c_1, \dots, c_N соответственно.

Ими наполняют рюкзак, который выдерживает вес не более M . Какую наибольшую стоимость могут иметь предметы в рюкзаке?

В первой строке вводится натуральное число N , не превышающее 100 и натуральное число M , не превышающее 10000.

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

В третьей строке вводятся N натуральных чисел c_i , не превышающих 100.

Выведите одно целое число: наибольшую возможную стоимость рюкзака.

Input	Output
4 6 2 4 1 2 7 2 5 1	13

Сформулируйте различные жадные стратегии для этой задачи и приведите контрпример для каждой из них.

Ф. *0-1 рюкзак — максимальная стоимость с восстановлением ответа*

Дано N предметов массой m_1, \dots, m_N и стоимостью c_1, \dots, c_N соответственно.

Ими наполняют рюкзак, который выдерживает вес не более M . Определите набор предметов, который можно унести в рюкзаке, имеющий наибольшую стоимость.

В первой строке вводится натуральное число N , не превышающее 100 и натуральное число M , не превышающее 10000.

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

В третьей строке вводятся N натуральных чисел c_i , не превышающих 100.

Выведите номера предметов (числа от 1 до N), которые войдут в рюкзак наибольшей стоимости.

Input	Output
4 6 2 4 1 2 7 2 5 1	1 3 4

Г. *0-1 рюкзак — точный вес, минимум предметов*

Дано N предметов массой m_1, \dots, m_N .

Ими наполняют рюкзак, который выдерживает вес не более M . Как набрать вес в точности M , используя как можно меньше предметов?

Первая строка входных данных содержит натуральное число N , не превышающее 100 и натуральное число M , не превышающее 10000.

Во второй строке находится N натуральных чисел m_i , не превышающих 100.

Выведите наименьшее необходимое число предметов или 0, если набрать данный вес невозможно.

Input	Output
4 6 4 2 3 1	2

Н. *Гирьки — две кучки равной массы*

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

Первая строка входных данных содержит натуральное число N , не превышающее 100.

Далее идет N натуральных чисел m_i , не превышающих 100.

Программа должна вывести YES, если гирьки можно разложить на две кучки равной массы или NO в противном случае.

Input	Output
4 4 2 3 1	YES

И. *Гирьки — две кучки равной массы с восстановлением ответа*

Дан набор гирек массой m_1, \dots, m_N . Разделите этот набор на две кучки равной массы, содержащие равное число гирек.

Первая строка входных данных содержит натуральное число N , не превышающее 100.

Далее идет N натуральных чисел m_i , не превышающих 100.

Необходимо вывести в первой строчке номера гирек (числа от 1 до N), входящие в первую кучку, во второй строчке — номера гирек во второй кучке.

Если задача не имеет решения, выведите число 0.

Input	Output
4 4 2 3 1	2 3 1 4

Ж. *Гирьки — три кучки равной массы с восстановлением ответа*

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

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

идет N натуральных чисел m_i , не превышающих 60.

Программа должна вывести номера гирек для каждого из наборов в три строки или строчку No solution, если решения не существует.

Input	Output
5 4 2 3 1 5	5 1 4 2 3

К. *Гирьки — четыре кучки равной массы с восстановлением ответа*

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

Первая строка входных данных содержит натуральное число N , не превышающее 14. Далее

идет N натуральных чисел m_i , не превышающих 100.

Программа должна вывести номера гирек для каждого из наборов в три строки или строчку No solution, если решения не существует.

Input	Output
8 10 20 30 40 50 60 70 80	1 8 2 7 3 6 4 5

Л. *Гирьки — три кучки равного размера и равной массы*

Дан набор гирек массой m_1, \dots, m_N . Разделите его на три кучки равной массы, содержащие равное число гирек.

Первая строка входных данных содержит натуральное число N , не превышающее 18. Далее

идет N натуральных чисел m_i , не превышающих 100.

Программа должна вывести номера гирек для каждого из наборов в три строки или строчку No solution, если решения не существует.

Input	Output
6 10 20 30 40 50 60	1 6 2 5 3 4