

Задача 01. Двоичные коды

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

Сгенерировать все двоичные вектора длины n .

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 16$).

Формат выходных данных

Выведите в лексикографическом порядке все двоичные вектора длины n .

Пример

стандартный ввод	стандартный вывод
3	000 001 010 011 100 101 110 111

Задача 02. Коды Грея для двоичных векторов

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

Сгенерировать двоичный код Грея длины n .

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 16$).

Формат выходных данных

Выведите в порядке произвольного кода Грея все двоичные вектора длины n .

Пример

стандартный ввод	стандартный вывод
3	000 001 011 010 110 111 101 100

Задача 03. Коды Антигрея

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

Сгенерировать троичный код Антигрея длины n .

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 10$).

Формат выходных данных

Выведите все троичные вектора длины n , так чтобы в соседних отличались значения на всех n позициях.

Пример

стандартный ввод	стандартный вывод
2	00 11 22 10 21 02 20 01 12

Задача 04. Цепной код

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

Сгенерировать двоичные вектора длины n , в порядке какого-нибудь цепного кода.

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 15$).

Формат выходных данных

Выведите все двоичные вектора длины n , в порядке какого-нибудь цепного кода.

Пример

стандартный ввод	стандартный вывод
3	000 001 010 101 011 111 110 100

Задача 05. Телеметрия

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

Сгенерируйте k -ичные вектора длины n , так чтобы у двух подряд идущих векторов, значения на всех кроме одной позиции совпадали, а значения на оставшейся позиции отличались ровно на 1.

Формат входных данных

В первой строке ввода дано два целых числа n и k ($n \geq 2$, $2 \leq k \leq 9$, $1 \leq k^n \leq 100000$).

Формат выходных данных

Выведите все k -ичные вектора длины n , так чтобы у двух подряд идущих векторов, значения на всех кроме одной позиции совпадали, а значения на оставшейся позиции отличались ровно на 1.

Пример

стандартный ввод	стандартный вывод
2 2	00 10 11 01

Задача 06. Двоичные вектора

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

Сгенерировать двоичные вектора длины n , в которых нет двух единиц подряд.

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 16$).

Формат выходных данных

В первой строке выведите количество двоичных векторов длины n в которых нет двух единиц подряд. В следующих строках выведите сами эти вектора в лексикографическом порядке по одному в строке.

Пример

стандартный ввод	стандартный вывод
4	8 0000 0001 0010 0100 0101 1000 1001 1010

Задача 07. Перестановки

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

Сгенерируйте перестановки длины n .

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 9$).

Формат выходных данных

Выведите в лексикографическом порядке все перестановки чисел от 1 до n .

Пример

стандартный ввод	стандартный вывод
3	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1

Задача 08. Сочетания

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

Сгенерируйте сочетания из n по k .

Формат входных данных

В первой строке ввода дано два целых числа n и k ($1 \leq k \leq n \leq 16$).

Формат выходных данных

Выведите все сочетания по k из чисел от 1 до n в лексикографическом порядке.

Пример

стандартный ввод	стандартный вывод
4 2	1 2 1 3 1 4 2 3 2 4 3 4

Задача 09. Правильные скобочные последовательности

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

Сгенерируйте правильные скобочные последовательности длины $2n$.

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 10$).

Формат выходных данных

Выведите все правильные скобочные последовательности с n открывающимися скобками в лексикографическом порядке, «(» < «)».

Пример

стандартный ввод	стандартный вывод
4	((((())) (((())) ((() ()) ((()) () (() (()) (() () () (() ()) () (() () ()) (() () ()) (() () ()) (() ((())) (() (() ()) (() (()) () (() (()) ()) (() (()) ())

Задача 10. Разбиения на слагаемые

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

Сгенерировать разбиения на слагаемые числа n .

Формат входных данных

В первой строке ввода дано одно целое число n ($2 \leq n \leq 40$).

Формат выходных данных

Выведите все разбиения числа n на слагаемые по одному в строке. Слагаемые следует выводить в возрастающем порядке. Разбиения отличающиеся только порядком слагаемых считаются одинаковыми.

Пример

стандартный ввод	стандартный вывод
4	1+1+1+1 1+1+2 1+3 2+2 4

Задача 11. Подмножества

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

Сгенерируйте подмножества множества $\{1, 2, \dots, n\}$.

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 10$).

Формат выходных данных

Выведите все подмножества множества $\{1, 2, \dots, n\}$ в лексикографическом порядке.

Пример

стандартный ввод	стандартный вывод
3	1 1 2 1 2 3 1 3 2 2 3 3

Задача 12. Разбиения на множества

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

Сгенерируйте разбиения n -элементного множества на k неупорядоченных множеств

Формат входных данных

В первой строке ввода дано два целых числа n и k ($1 \leq k \leq n \leq 10$).

Формат выходных данных

Выведите все разбиения n -элементного множества на k неупорядоченных множеств. Разбиения можно выводить в любом порядке. Внутри разбиения множества можно выводить в любом порядке. Внутри множества числа надо выводить в возрастающем порядке. Следуйте формату из примера.

Пример

стандартный ввод	стандартный вывод
4 2	1 2 3 4 1 2 3 4 1 3 2 4 1 2 3 4 1 4 2 3 1 2 4 3 1 3 4 2

Задача 13. Перестановка по номеру

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

Сгенерируйте k -ю в лексикографическом порядке перестановку чисел от 1 до n .

Формат входных данных

В первой строке ввода дано два целых числа n ($1 \leq n \leq 18$, $0 \leq k \leq n! - 1$).

Формат выходных данных

Выведите k -ю в лексикографическом порядке перестановку чисел от 1 до n . Перестановки занумерованы от 0 до $n! - 1$.

Пример

стандартный ввод	стандартный вывод
3 4	3 1 2

Задача 14. Номер по перестановке

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

По перестановке длины n определите её номер. Перестановки занумерованы, начиная с 0.

Формат входных данных

В первой строке ввода дано одно целое число n ($1 \leq n \leq 18$). Во второй строке ввода дана перестановка чисел от 1 до n .

Формат выходных данных

Выведите номер заданной перестановки в лексикографическом порядке всех перестановок чисел от 1 до n .

Пример

стандартный ввод	стандартный вывод
3 3 1 2	4

Задача 15. Сочетание по номеру

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

Сгенерируйте m -е сочетание из n по k по его номеру.

Формат входных данных

В первой строке ввода даны три целых числа n , k и m ($1 \leq k \leq n \leq 30$, $0 \leq m \leq \binom{n}{k} - 1$).

Формат выходных данных

Выведите m -е в лексикографическом порядке сочетание по k из чисел от 1 до n . Сочетания занумерованы, начиная с 0.

Пример

стандартный ввод	стандартный вывод
4 2 3	2 3

Задача 16. Номер по сочетанию

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

Определите номер сочетания из n по k .

Формат входных данных

В первой строке дано два целых числа n и k ($1 \leq k \leq n \leq 30$). Во второй строке дано сочетание, состоящее из k чисел от 1 до n .

Формат выходных данных

Выведите в выходной файл номер этого сочетания в лексикографическом порядке всех сочетаний из n чисел по k . Сочетания нумеруются, начиная с 0.

Пример

стандартный ввод	стандартный вывод
4 2 2 3	3

Задача 17. Правильная скобочная последовательность по номеру

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

Сгенерируйте k -ю в лексикографическом порядке правильную скобочную последовательность длины $2n$.

Формат входных данных

В первой строке ввода дано два целых числа n и k ($1 \leq n \leq 20$).

Формат выходных данных

Выведите k -ю в лексикографическом порядке правильную скобочную последовательность среди всех правильных скобочных последовательностей с n открывающимися скобками, упорядоченных в лексикографическом порядке, «(» < «)». Последовательности занумерованы, начиная с 0. Искомая последовательность существует.

Пример

стандартный ввод	стандартный вывод
4 3	((()))()

Задача 18. Номер по правильной скобочной последовательности

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

Определите номер правильной скобочной последовательности.

Формат входных данных

В первой строке задана правильная скобочная последовательность. Количество открывающихся скобок в последовательности — от 1 до 20.

Формат выходных данных

Выведите ее номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, «(» < «)». Последовательности занумерованы, начиная с 0.

Пример

стандартный ввод	стандартный вывод
((()))()	3

Задача 19. ПСП с двумя типами скобок по номеру

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

Сгенерируйте k -ю в лексикографическом порядке правильную скобочную последовательность длины $2n$ с двумя типами скобок.

Формат входных данных

В первой строке ввода дано два целых числа n и k ($1 \leq n \leq 20$).

Формат выходных данных

Выведите в выходной файл k -ю в лексикографическом порядке правильную скобочную последовательность среди всех правильных скобочных последовательностей с двумя типами скобок с n открывающимися скобками, упорядоченных в лексикографическом порядке, «(» < «)» < «[» < «]». Последовательности занумерованы, начиная с 0. Искомая последовательность существует.

Пример

стандартный ввод	стандартный вывод
4 100	([]) () []

Задача 20. Номер по ПСП с двумя типами скобок

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

Определите номер правильной скобочной последовательности с двумя типами скобок.

Формат входных данных

В первой строке задана правильная скобочная последовательность с двумя типами скобок. Количество открывающихся скобок в последовательности — от 1 до 20.

Формат выходных данных

Выведите ее номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, «(» < «)» < «[» < «]». Последовательности занумерованы, начиная с 0.

Пример

стандартный ввод	стандартный вывод
([]) () []	100

Задача 21. Разбиение на слагаемые по номеру

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

Рассмотрим все разбиения числа n на слагаемые, в каждом разбиении упорядочим их в порядке не убывания. Будем считать, что разбиение $a_1 + a_2 + \dots + a_n$ лексикографически меньше $b_1 + b_2 + \dots + b_m$, если для некоторого $k \forall j \leq k : a_j = b_j$ и либо $k = n$, либо $a_{k+1} < b_{k+1}$.

Формат входных данных

Во вводе заданы числа n и r . $1 \leq n \leq 100$, разбиение с номером r — существует.

Формат выходных данных

Выведите r -ое разбиение числа n на слагаемые, разбиения нумеруются с 0.

Пример

стандартный ввод	стандартный вывод
4 3	2+2

Задача 22. Номер по разбиению на слагаемые

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

Рассмотрим все разбиения числа n на слагаемые, в каждом разбиении упорядочим их в порядке не убывания. Будем считать, что разбиение $a_1 + a_2 + \dots + a_n$ лексикографически меньше $b_1 + b_2 + \dots + b_m$, если для некоторого $k \forall j \leq k : a_j = b_j$ и либо $k = n$, либо $a_{k+1} < b_{k+1}$.

Формат входных данных

Во вводе задано разбиение на слагаемые.

Формат выходных данных

Выведите номер этого разбиения, среди всех разбиений упорядоченных лексикографически. Разбиения нумеруются с 0. Гарантируется, что в разбиении слагаемые упорядочены в порядке не убывания, и $1 \leq n \leq 100$.

Пример

стандартный ввод	стандартный вывод
2+2	3

Задача 23. Предыдущий и следующий двоичный вектор

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

По двоичному вектору сгенерируйте предыдущий и следующий двоичный вектор.

Формат входных данных

Во вводе задан двоичный вектор. Длина вектора во входном файле — от 1 до 200000.

Формат выходных данных

Выведите предыдущий и следующий двоичный вектор в лексикографическом порядке. Если какого-либо из них не существует, выведите вместо него «-».

Примеры

стандартный ввод	стандартный вывод
10001	10000 10010
0	- 1

Задача 24. Предыдущая и следующая перестановки

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

По перестановке сгенерируйте предыдущую и следующую перестановку чисел от 1 до n .

Формат входных данных

Во вводе задано число n и затем перестановка чисел от 1 до n ($1 \leq n \leq 100\,000$).

Формат выходных данных

Выведите предыдущую и следующую перестановку чисел от 1 до n . Если какой либо из них не существует, выведите вместо нее n нулей.

Примеры

стандартный ввод	стандартный вывод
4 1 3 2 4	1 2 4 3 1 3 4 2
2 1 2	0 0 2 1

Задача 25. Следующее сочетание

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

По сочетанию из n по k сгенерируйте следующее сочетание.

Формат входных данных

Во вводе заданы числа n , k и затем сочетание, состоящее из k чисел от 1 до n ($1 \leq k \leq n \leq 10000$).

Формат выходных данных

Выведите следующее сочетание в лексикографическом порядке из n чисел по k .

Если его не существует, выведите -1.

Пример

стандартный ввод	стандартный вывод
4 2 2 3	2 4

Задача 26. Следующее разбиение на множества

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

Рассмотрим множество первых n натуральных чисел: $N_n = \{1, 2, \dots, n\}$. Разбиением на множества называется представление этого множества, как объединения одного или более, попарно непересекающихся подмножеств множеств. Например для $n = 5$ существуют следующие разбиения:

$$\begin{aligned}\{1, 2, 3, 4, 5\} &= \{1, 2, 3\} \cup \{4, 5\} \\ \{1, 2, 3, 4, 5\} &= \{1, 3, 5\} \cup \{2, 4\} \\ \{1, 2, 3, 4, 5\} &= \{1, 2, 3, 4, 5\} \\ \{1, 2, 3, 4, 5\} &= \{1\} \cup \{2\} \cup \{3\} \cup \{4\} \cup \{5\}\end{aligned}$$

Всего существует 52 разбиения множества N_5 . Заметьте, что мы не различаем разбиения на множества, которые отличаются только порядком подмножеств.

Упорядочим все разбиения на множества N_n лексикографически. Для этого во-первых в каждом разбиении упорядочим множества лексикографически. Будем говорить, что подмножество $A \subset N_n$ лексикографически меньше подмножества $B \subset N_n$, если верно одно из следующих условий:

- существует i такое, что $i \in A$, $i \notin B$, для всех $j < i$: $j \in A$ если и только если $j \in B$, и существует $k > i$ такое что $k \in B$;
- $A \subset B$ и $i < j$ для всех $i \in A$ и $j \in B \setminus A$.

Разбиения упорядочены лексикографически следующим образом. Разбиение $N_n = A_1 \cup A_2 \cup \dots \cup A_k$ лексикографически меньше разбиения $N_n = B_1 \cup B_2 \cup \dots \cup B_l$ если существует такое i , что $A_1 = B_1$, $A_2 = B_2$, \dots , $A_{i-1} = B_{i-1}$ и $A_i < B_i$.

Дано разбиение N_n , ваша задача найти следующее разбиение на множества в лексикографическом порядке.

Формат входных данных

Во вводе содержится несколько тестов. Каждый тест в первой строчке содержит n и k — количество чисел в разбиваемом множестве, и количество подмножеств в разбиении. ($1 \leq n \leq 200$). Следующие k строк содержат элементы разбиения. Элементы в каждом подмножестве упорядочены по возрастанию.

Тесты разделены пустой строкой. Последняя строка содержит два нуля.

Сумма всех n по всем тестам не превосходит 2000.

Формат выходных данных

Для каждого теста выведите следующее разбиение. Если разбиение вводе является последним в лексикографическом порядке, то выведите первое в лексикографическом порядке разбиение. Используйте такой же формат, как и во вводе. Разделяйте ответы для разных тестов пустой строкой.

Пример

стандартный ввод	стандартный вывод
5 2 1 2 3 4 5	5 2 1 2 3 4 5
5 2 1 3 5 2 4	5 4 1 4 2 3 5
5 1 1 2 3 4 5	5 2 1 2 3 5 4
5 5 1 2 3 4 5	5 4 1 2 3 4 5
0 0	

Задача 27. Следующая правильная скобочная последовательность

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

По ПСП сгенерируйте следующее ПСП.

Формат входных данных

Во вводе задана правильная скобочная последовательность. Количество открывающихся скобок в последовательности — от 1 до 100 000.

Формат выходных данных

Выведите следующую за ней в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, «(» < «)». Если такой нет, выведите «-».

Пример

стандартный ввод	стандартный вывод
((()))()	((())())

Задача 28. Следующая мультиперестановка

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

По мультиперестановке сгенерируйте следующую мультиперестановку.

Формат входных данных

Во вводе задано число n и затем мультиперестановка, составленная из чисел от 1 до n ($1 \leq n \leq 100\,000$).

Формат выходных данных

Выведите следующую в лексикографическом порядке мультиперестановку того же мультимножества. Если искомой перестановки не существует, выведите n нулей.

Пример

стандартный ввод	стандартный вывод
4 1 3 2 4	1 3 4 2

Задача 29. Следующее разбиение на слагаемые

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

Разбиения числа n на слагаемые — это набор целых положительных чисел, сумма которых равна n . При этом разбиения, отличающиеся лишь порядком слагаемых, считаются одинаковыми, поэтому можно считать, что слагаемые в разбиении упорядочены по неубыванию.

Например, существует 7 разбиений числа 5 на слагаемые:

$$5 = 1 + 1 + 1 + 1 + 1$$

$$5 = 1 + 1 + 1 + 2$$

$$5 = 1 + 1 + 3$$

$$5 = 1 + 2 + 2$$

$$5 = 1 + 4$$

$$5 = 2 + 3$$

$$5 = 5$$

В приведенном примере разбиения упорядочены лексикографически — сначала по первому слагаемому в разбиении, затем по второму, и так далее. В этой задаче вам потребуется по заданному разбиению на слагаемые найти следующее в лексикографическом порядке разбиение.

Формат входных данных

Ввод содержит одну строку — разбиение числа n на слагаемые ($1 \leq n \leq 100\,000$). Слагаемые в разбиении следуют в неубывающем порядке.

Формат выходных данных

Выведите одну строку — разбиение числа n на слагаемые, следующее в лексикографическом порядке после приведенного во входном файле. Если во входном файле приведено последнее разбиение числа n на слагаемые, выведите «No solution».

Примеры

стандартный ввод	стандартный вывод
5=1+1+3	5=1+2+2
5=5	No solution