

Задача А. Еще одна строковая задача

Имя входного файла: `stdin`
 Имя выходного файла: `stdout`
 Ограничение по времени: 4 секунды
 Ограничение по памяти: 256 мегабайт

Строка называется бинарной, если она состоит только из символов «0» и «1».

Строка v называется подстрокой строки w , если она имеет ненулевую длину, и ее можно прочитать, начиная с некоторой позиции, в строке w . Например, у строки «010» есть шесть подстрок: «0», «1», «0», «01», «10», «010». Две подстроки считаются различными, если их позиции вхождения различны. Другими словами, каждую подстроку нужно учитывать столько раз, сколько она встречается.

Дана бинарная строка s . Ваша задача — найти количество ее подстрок, содержащих ровно k единиц.

Формат входного файла

В первой строке записано единственное целое число k ($0 \leq k \leq 10^6$). Во второй строке записана непустая бинарная строка s . Длина s не превосходит 10^6 символов.

Формат выходного файла

Выведите одно целое число — количество подстрок данной строки, содержащих ровно k символов «1».

Примеры

stdin	stdout
1 1010	6

Задача В. Тест на тупость

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

У Саши есть n палочек с длинами a_1, \dots, a_n . Ваша задача — определить количество различных тупоугольных треугольников, которые можно составить из них. Каждый треугольник должен быть составлен ровно из трёх палочек набора.

Формат входного файла

Входной файл содержит n строк, каждая из которых содержит очередное число a_i . Все числа натуральные, все не превышают 7 000.

Формат выходного файла

Выведите строку «`Sasha can compose k triangles`», где k — количество тупоугольных треугольников.

Примеры

	obtuse.in	obtuse.out
3		Sasha can compose 2 triangles
4		
5		
5		
5		
6		

Задача С. Тест на тупость: 2

Имя входного файла: `sum2.in`
 Имя выходного файла: `sum2.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 256 мегабайт

У Александра есть n палочек длинами a_1, \dots, a_n . И ещё n палочек длинами b_1, \dots, b_n . Ваша задача — определить, может ли Саша из одной палочки первого набора и одной палочки второго набора составить палочку длиной s .

Формат входного файла

В первой строке записаны числа n и s — количество чисел в наборах и требуемая длина. Во второй и третьей строке записано по n натуральных чисел: наборы a и b . Наборы отсортированы по неубыванию. $n \leq 100\,000$. Все числа — натуральные, ни одно не превышает 10^9 .

Формат выходного файла

Выведите пару чисел i, j , если палочки a_i и b_j складываются в палочку длины s . Если такой пары палочек нет, выведите одну пару нулей.

Примеры

	sum2.in	sum2.out
3	7	1 2
2	3 9	
1	5 7	

Задача D. Тест на тупость: 4

Имя входного файла: `sum4.in`
 Имя выходного файла: `sum4.out`
 Ограничение по времени: 12 секунды
 Ограничение по памяти: 8 мегабайт

У Александры есть n палочек длинами a_1, \dots, a_n . И ещё n палочек длинами b_1, \dots, b_n . А также n палочек длинами c_1, \dots, c_n . И последние n палочек длинами d_1, \dots, d_n . Ваша задача — определить, может ли Саша из одной палочки первого набора, одной палочки второго набора, одной палочки третьего набора и одной палочки четвертого набора составить палочку длиной s .

Формат входного файла

В первой строке записаны числа n и s — количество чисел в наборах и требуемая длина. Во второй, третьей, четвёртой и пятой строках записано по n натуральных чисел: наборы a , b , c и d . Наборы отсортированы по неубыванию. $n \leq 5000$. Все числа — натуральные, ни одно не превышает 10^9 .

Формат выходного файла

Выведите четвёрку чисел i, j, k, l , если палочки a_i, b_j, c_k, d_l складываются в палочку длины s . Если такой четвёрки палочек нет, выведите одну четвёрку нулей.

Примеры

sum4.in	sum4.out
3 10	1 2 3 3
2 3 9	
1 5 7	
1 1 1	
2 2 2	

Примеры

dowry.in	dowry.out
3 6 8	1
3 10	2
7 3	
8 2	

Задача Е. Сокровища

Имя входного файла: dowry.in
Имя выходного файла: dowry.out
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 входного файла

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 выходного файла

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

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

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