

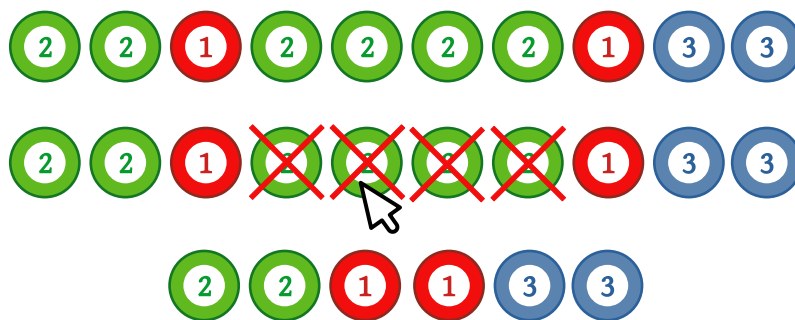
Задача 1. Lines 2020

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Известная марсианская компания «Interplanetary Software, Inc.» разрабатывает новую мобильную игру — Lines 2020.

В Lines 2020 игровое поле представляет собой n разноцветных шаров, расположенных в ряд, причем i -й шар имеет цвет c_i . За один ход игрок может выбрать какой-либо шар и уничтожить его. При этом если сосед или соседи этого шара имели такой же цвет, то они также уничтожатся, далее уничтожатся соседи соседей, имеющие такой же цвет, и так далее. После этого игровое поле «схлопнется», то есть шары сдвинутся вместе, таким образом, чтобы снова образовывать непрерывную линию.

В качестве примера рассмотрим рисунок ниже.



Если уничтожить какой-либо из шаров цвета 2 посередине, то уничтожится вся группа из четырех соседних шаров. После этого последовательность цветов станет такой: (2, 2, 1, 1, 3, 3).

Стоит также упомянуть, что по правилам игры последовательность не может остаться пустой. Как только игрок не может сделать ход (поскольку любой ход приведет к уничтожению всех шаров на поле), то тогда игра заканчивается, и *балансом игры* называется **сумма номеров цветов** на оставшихся шарах. Чем меньше баланс игры, тем лучше.

Чтобы качественно разработать уровни для этой игры, компании необходимо по заданной расстановке уметь рассчитывать минимально возможный финальный баланс игры. Напишите программу для вычисления минимального баланса по начальному состоянию игрового поля.

Формат входных данных

В первой строке входных данных находится целое число n ($1 \leq n \leq 10^5$) — количество шаров на игровом поле.

Во второй строке входных данных находится n целых чисел a_i ($1 \leq a_i \leq 10^9$) — цвет i -го шара.

Формат выходных данных

Выведите целое число s — минимально возможный баланс игры.

Система оценки

№	Дополнительные ограничения	Баллы за подзадачу
1	$n \leq 10$	13
2	$n \leq 20$	18
3	$n \leq 100$	20
4	Нет дополнительных ограничений	49

Примеры

input.txt	output.txt
6 2 2 1 1 1 3	3
5 1 1 1 1 1	5
6 9 4 2 8 5 7	2

Замечание

В первом примере можно уничтожить сначала группу шаров цвета 2 слева, а затем шар цвета 3 справа. Дальше уничтожать шары нельзя, поскольку иначе поле станет пустым, что недопустимо. Так как осталось три шара цвета 1, то ответ равен $1 + 1 + 1 = 3$.

Во втором примере ничего уничтожить нельзя, поэтому ответ равен $1 + 1 + 1 + 1 + 1 = 5$.

В третьем примере оптимально оставить только шар цвета 2, поэтому ответ равен двум.

Задача 2. Радиотелескопы

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Как Вы уже, наверное, знаете, марсиане активно занимаются исследованием Вселенной. Для этого был построен крупный научный центр — Институт Исследования Космоса (ИИК).

Около ИИК установлено в ряд n радиотелескопов, которые получают излучение от далеких планет и звезд. Каждый телескоп характеризуется своей мощностью a_i .

Известно, что стоящие рядом телескопы взаимодействуют между собой с некоторой силой. Для телескопов номер i и $i + 1$ эта сила взаимодействия равна $a_i \oplus a_{i+1}$, где \oplus обозначает операцию побитового исключающего ИЛИ (см. Замечание). Мощность всей системы телескопов определяется суммарной силой взаимодействия между соседними телескопами. Иными словами, мощность всей системы S равна

$$S = (a_1 \oplus a_2) + (a_2 \oplus a_3) + \dots + (a_{n-1} \oplus a_n).$$

К сожалению, на Марсе наступил экономический кризис, и финансирование космических исследований пришлось сократить. Поскольку телескопы довольно дороги в обслуживании, то ИИК может оставить не более k из них. При этом мощность системы телескопов должна быть максимально возможной. При этом оставшиеся телескопы сохраняют свой относительный порядок, то есть менять их местами запрещено. Иными словами, требуется выбрать t ($1 \leq t \leq k$) индексов $1 \leq i_1 < i_2 < \dots < i_t \leq n$ таких, что мощность системы телескопов $(a_{i_1}, a_{i_2}, \dots, a_{i_t})$ максимально возможная.

Решите эту задачу и определите способ выбрать телескопы оптимальным образом.

Обратите внимание, что в каждой из подзадач накладываются какие-либо дополнительные ограничения на входные данные.

Формат входных данных

В первой строке входных данных находится два целых числа n и k ($2 \leq k \leq n \leq 2 \cdot 10^5$) — количество радиотелескопов в ИИК и максимальное количество телескопов, которое можно оставить.

Во второй строке входных данных находится n целых чисел a_i ($0 \leq a_i \leq 2^{30} - 1$) — мощность i -го телескопа.

Формат выходных данных

Выведите одно целое число — максимальная мощность системы, если оставить не более k телескопов.

Система оценки

№	Дополнительные ограничения	Баллы за подзадачу
1	$n \leq 20$	11
2	$n \leq 30$	24
3	$n \leq 300$	39
4	$k = n$	26

Примеры

input.txt	output.txt
3 3 3 7 6	5
3 2 3 7 6	5
6 3 1 4 2 8 5 7	27
6 4 8 8 8 1 3 6	16

Замечание

Исключающее ИЛИ — операция, которая на вход принимает два бита и возвращает бит 0, если биты на входе равны и 1 в противном случае. Например: $1 \oplus 0 = 1$, $1 \oplus 1 = 0$. Для применения побитового исключающего ИЛИ двух чисел эти числа сначала переводят в двоичную систему счисления, а затем применяют исключающее ИЛИ к каждому из разрядов. Например, $6 \oplus 3 = 5$, поскольку $6 = 110_2$, $3 = 11_2$. Применив исключающее ИЛИ поразрядно, получаем $5 = 101_2$:

$$\begin{array}{r} 110_2 \\ \oplus 11_2 \\ \hline 101_2 \end{array}$$

В языке программирования Pascal побитовое исключающее ИЛИ чисел a и b обозначается $a \text{ xor } b$, а в языках Python и C++ — $a \wedge b$.

Теперь рассмотрим примеры.

В первых двух примерах можно оставить два телескопа с мощностью 3 и 6. Тогда итоговая мощность системы телескопов будет равна $3 \oplus 6 = 5$.

В третьем примере оптимально взять телескопы с мощностями 4, 8 и 7. Итоговая мощность будет равна $(4 \oplus 8) + (8 \oplus 7) = 12 + 15 = 27$.

В четвертом примере можно оставить телескопы с мощностями 8, 1, 3 и 6 и получить итоговую мощность $(8 \oplus 1) + (1 \oplus 3) + (3 \oplus 6) = 9 + 2 + 5 = 16$. Обратите внимание, что менять относительный порядок телескопов запрещено, иначе возможно было бы переставить телескопы в порядке 8, 6, 3, 1 и получить ответ $(8 \oplus 6) + (6 \oplus 3) + (3 \oplus 1) = 14 + 5 + 2 = 21$.

Задача 3. Добыча меди

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Кидония — область на Марсе, которая славится своими богатыми залежами медной руды. В этой области находится n городов, выстроенных вдоль одной прямой дороги, идущей с запада на восток. Города пронумерованы целыми числами от 1 до n в порядке следования вдоль дороги. Около каждого города располагаются рудники, около i -го города находится a_i рудников.

Недавно марсиане заключили крупный контракт на поставку меди на Венеру, поэтому им требуется увеличить эффективность добычи этого металла. Но прежде всего необходимо решить проблему управления рудниками Кидонии и разбить все города на районы.

Правительство решило произвести разбиение на районы следующим образом. Сначала все города помещаются в один большой район. Далее происходит следующее:

- либо район считается конечным, т. е. не разбивается на более мелкие районы (такой район называется *терминальным*);
- либо район разбивается на два непустых района (причем города в одном районе не обязаны идти последовательно). Полученные районы могут быть рекурсивно разбиты на более мелкие по аналогичной процедуре.

После разбиения на районы можно посчитать их эффективность следующим образом:

- Эффективность терминального района равна среднему количеству рудников во всех городах этого района. Иными словами, если в районе k городов с количеством рудников b_1, b_2, \dots, b_k , то эффективность района равна $\frac{b_1+b_2+\dots+b_k}{k}$.
- Если же район не является терминальным, то его эффективность равна средней эффективности двух районов, на которые он разбивается, т. е. если он разбивается на районы с эффективностью x и y соответственно, то тогда эффективность данного района будет равна $\frac{x+y}{2}$.

Чтобы добыча меди была максимально эффективной, руководство Кидонии хочет максимизировать эффективность самого большого района (содержащего все города).

Поскольку проект по повышению эффективности пока что является экспериментальным, власти решили разбивать на районы не все города Кидонии, а только те, которые имеют номера с l по r включительно. То есть, самый большой район будет содержать только города с номерами l по r .

Какой именно отрезок городов использовать для экспериментов, правительство Кидонии пока не решило. Было утверждено q возможных вариантов. А Вам было поручено для каждого из q вариантов рассчитать максимально возможную эффективность при оптимальном делении на районы. Справитесь?

Формат входных данных

В первой строке входных данных находится целое число n ($1 \leq n \leq 10^5$) — количество городов в Кидонии.

Во второй строке входных данных находится n целых чисел a_i ($1 \leq a_i \leq 10^6$) — количество рудников в i -м городе.

В третьей строке входных данных расположено целое число q ($1 \leq q \leq 10^5$) — количество вариантов, которые необходимо рассмотреть.

В каждой из следующих q строк находится по два целых числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$). Это означает, что в i -м варианте требуется разбивать на районы лишь те города, которые имеют номера с l_i по r_i включительно.

Формат выходных данных

Выведите q строк. В i -й строке выведите вещественное число e_i — максимально возможную эффективность разбиения на районы всех городов с номерами от l_i до r_i .

Выводить ответ следует с абсолютной погрешностью не более 10^{-4} . Иными словами, если ответ жюри для i -го варианта равен o_i , а Ваш ответ равен e_i , то он считается правильным, если $|o_i - e_i| \leq 10^{-4}$.

Система оценки

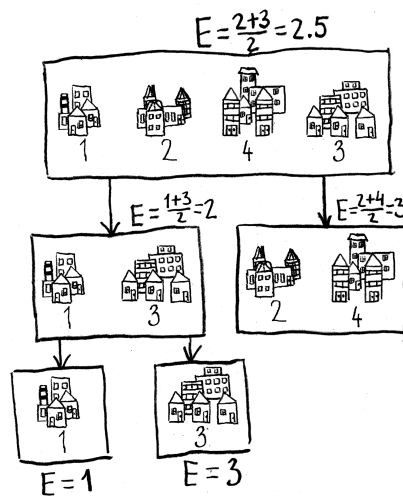
№	Дополнительные ограничения	Баллы за подзадачу
1	$n \leq 15, q = 1$	18
2	$n \leq 3000, q = 1$	9
3	$q = 1$	19
4	$a_i \leq 3$	26
5	Нет дополнительных ограничений	28

Пример

input.txt	output.txt
7	3.1250000000
2 1 2 4 3 3 3	3.0000000000
3	1.7500000000
2 5	
5 7	
1 3	

Замечание

В первом запросе требуется разбить на районы города $\{1, 2, 4, 3\}$. Одно из разбиений может выглядеть следующим образом:



Как видно из рисунка, эффективность такого разбиения равна 2.5. Однако оптимальным является разбиение с эффективностью 3.125.

Во втором запросе необходимо разбить на районы города $\{3, 3, 3\}$. Здесь оптимальным является любое разбиение, в частности, можно вообще не проводить разбиение на более мелкие районы, и эффективность такого разбиения будет равна $\frac{3+3+3}{3} = 3$.

В третьем запросе можно разбить города на два района, $\{1, 2\}$ (эффективность 1.5) и $\{2\}$ (эффективность 2). Тогда эффективность всех городов будет равна $\frac{1.5+2}{2} = 1.75$.

Задача 4. Космическая реклама

Имя входного файла: `inputX.txt`
Имя выходного файла: `outputX.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Крупная межпланетная сеть супермаркетов «Galaxy-Wide Shopping, Inc.» начинает активную рекламную кампанию. В качестве рекламного слогана было решено использовать строку s , состоящую из n символов.

Недавно было проведено исследование, согласно которому чем больше в рекламном слогане подстрок, являющихся палиндромами, тем более запоминающимся он получится. Напоминаем, что палиндромом называется строка, которая читается одинаково в обоих направлениях. Например, строки «`abacaba`» и «`olympylo`» — палиндромы, а строки «`mars`» и «`abaa`» — нет.

Будем считать, что *запоминаемость* строки s равна числу подстрок-палиндромов. Например, у строки «`aba`» всего шесть подстрок: «`a`», «`b`», «`a`», «`ab`», «`ba`» и «`aba`», из них четыре («`a`», «`b`», «`a`» и «`aba`») являются палиндромами. Следовательно, запоминаемость строки «`aba`» равна четырем. Обратите внимание, что если подстрока встречается несколько раз, то каждое вхождение учитывается отдельно.

К сожалению, сотрудники рекламного отдела компании узнали об этом исследовании слишком поздно, когда все рекламные плакаты уже были напечатаны. Теперь они могут изменить не более k символов в рекламном слогане, чтобы его запоминаемость стала как можно больше. Сотрудники компании обратились за помощью к Вам, пообещав заплатить тем больше, чем больше станет запоминаемость рекламного слогана после исправления.

Формат входных данных

Входные данные находятся в файлах `input1.txt`, `input2.txt`, ..., `input10.txt`.

Первая строка входных данных содержит целое число t — номер теста (для примера из условия $t = 0$).

Вторая строка входных данных содержит два целых числа n и k ($1 \leq k \leq n$) — длина строки и максимальное количество символов, которое разрешается исправить.

В третьей строке входных данных находится строка s , состоящая из n строчных букв латинского алфавита — исходный рекламный слоган.

Формат выходных данных

На проверку необходимо сдать выходные файлы с названиями `output1.txt`, `output2.txt`, ..., `output10.txt`, где выходной файл `outputX.txt` должен соответствовать входному файлу `inputX.txt`.

Выведите строку s_1 длины n — рекламный слоган после всех исправлений. Строка s_1 должна получаться из строки s путем изменения одного символа строки на другой не более k раз.

Система оценки

Если выходной файл не соответствует указанному формату выходных данных, то Вы получите 0 баллов за тест.

Иначе Ваш балл за тест будет равен $\min\left(10, 10 \cdot \left(\frac{out}{ans}\right)^2\right)$, где out — запоминаемость найденного Вами рекламного слогана, а ans — запоминаемость рекламного слогана, найденного жюри.

Баллы за каждый тест округляются вверх до сотых и суммируются. Правила округления таковы, что, например, при округлении числа 10.112 вверх до сотых получаем число 10.12.

Пример

<code>inputX.txt</code>	<code>outputX.txt</code>
0	aaba
4 1	
aabc	

Замечание

Рассмотрим пример.

Путем замены последнего символа в строке на «а» мы получили строку «ааба». Ее запоминаемость равна 6, т. к. в ней содержится шесть палиндромов: «а» (три раза), «аа», «b» и «aba». Однако оптимальной строкой в этом примере является «ааас», которая получается заменой третьего символа исходной строки на «а». Запоминаемость строки «ааас» равна 7, поэтому за ответ «ааба» программа оценивания выставит $10 \cdot \left(\frac{6}{7}\right)^2 = 7.35$ балла.