

## Задача А. Красивый массив

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 256 мегабайт

Вам дан массив  $a$  состоящий из  $n$  целых чисел. Красота массива – это максимальная сумма какого-то последовательного подотрезка этого массива (этот подотрезок может быть пустым). Например красота массива  $[10, -5, 10, -4, 1]$  равна 15, а красота массива  $[-3, -5, -1]$  равна 0.

Вы можете выбрать не более одного последовательного подотрезка массива  $a$  и домножить все элементы этого подотрезка на  $x$ . Вы хотите максимизировать красоту массива после применения такой операции.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $x$ , ( $1 \leq n \leq 3 \cdot 10^5, -100 \leq x \leq 100$ ) — длина массива  $a$  и число  $x$  соответственно.

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$ , ( $-10^9 \leq a_i \leq 10^9$ ) — сам массив  $a$ .

### Формат выходных данных

Выведите одно число — максимальную красоту массива  $a$  после не более одного домножения непрерывного подотрезка этого массива на  $x$ .

### Примеры

стандартный ввод	стандартный вывод
5 -2 -3 8 -2 1 -6	22
12 -3 1 3 3 7 1 3 3 7 1 3 3 7	42
5 10 -1 -2 -3 -4 -5	0

## Задача В. Шоколадка

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

64 мегабайта

Команда «Отбой» участвует в очередном марафоне по «Угадай мелодию. Rock version». Чтобы было чем подкрепиться во время игры, команда взяла с собой большую прямоугольную плитку шоколада размерами  $w \times h$ . У команды есть список из  $n$  пар чисел — размеры шоколадок, которые команда считает счастливыми. Прежде чем приступить к поеданию шоколадки, участники команды решили поделить имеющуюся плитку на счастливые шоколадки. Для этого они действуют следующим образом: сначала плитка шоколада ломается на 2 части по линии, строго параллельной одной из своих сторон, после чего каждую из полученных частей они могут продолжить ломать аналогичным образом.

Вам поручили определить, какое максимальное количество счастливых шоколадок команда сможет получить, действуя по данной схеме. Шоколадки, полученные поворотом счастливых, счастливыми не являются.

### Формат входных данных

В первой строке входного файла заданы три целых числа  $w, h, n$  — размеры плитки шоколада и количество вариантов размера счастливых шоколадок соответственно ( $1 \leq w, h \leq 300, 1 \leq n \leq w \times h$ ). В следующих  $n$  строках заданы пары целых чисел  $w_i, h_i$  — размеры счастливых шоколадок ( $1 \leq w_i \leq w, 1 \leq h_i \leq h$ ).

### Формат выходных данных

В единственную строку выходного файла выведите максимальное количество счастливых шоколадок, на которые можно разрезать данную плитку.

### Примеры

стандартный ввод	стандартный вывод
21 11 4 10 4 6 2 7 5 15 10	15
9 12 5 1 12 2 6 3 4 4 3 6 2	9

## Задача С. Упаковка символов

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

Билл пытается компактно представить последовательности прописных символов от A до Z с помощью упаковки повторяющихся подпоследовательностей внутри них. Например, один из способов представить последовательность AAAAAAAAAABABABCCD - это 10(A)2(BA)B2(C) D. Он формально определяет сжатые последовательности символов и правила перевода их в несжатый вид следующим образом: Последовательность, содержащая один символ от A до Z, является упакованной. Распаковка этой последовательности даёт ту же последовательность из одного символа, повторённую  $\times$  раз. Следуя этим правилам, легко распаковать любую заданную упакованную последовательность. Однако Биллу более интересен обратный переход. Он хочет упаковать заданную последовательность так, чтобы результирующая сжатая последовательность содержала наименьшее возможное число символов. Ограничения: длина исходной последовательности от 1 до 100.

### Формат входных данных

В первой строке находится последовательность символов от A до Z.

### Формат выходных данных

В единственной строке выводится упакованная последовательность наименьшей длины, которая распаковывается в заданную последовательность. Если таких последовательностей несколько, выведите лексикографически минимальную.

### Примеры

стандартный ввод	стандартный вывод
AAAAAAAAAABABABCCD	9(A)3(AB)CCD
NEERCYESYESYESNEERCYESYESYES	2(NEERC3(YES))
A	A

## Задача D. Казино

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

Вновь открытое казино предложило оригинальную игру. В начале игры крупье выставляет в ряд несколько фишек разных цветов. Кроме того, он объявляет, какие последовательности фишек игрок может забирать себе в процессе игры. Далее игрок забирает себе одну из заранее объявленных последовательностей фишек, расположенных подряд. После этого крупье сдвигает оставшиеся фишки, убирая разрыв. Затем игрок снова забирает себе одну из объявленных последовательностей и так далее. Игра продолжается до тех пор, пока игрок может забирать фишки. Рассмотрим пример. Пусть на столе выставлен ряд фишек `rrrgggbbb`, и крупье объявил последовательности `rg` и `gb`. Игрок, например, может забрать фишки `rg`, лежащие на третьем и четвёртом местах слева. После этого крупье сдвинет фишки, и на столе получится ряд `rrggbbb`. Ещё дважды забрав фишки `rg`, игрок добьётся того, что на столе останутся фишки `bbb` и игра закончится, так как игроку больше нечего забрать со стола. Игрок мог бы действовать и по-другому - на втором и третьем ходах забрать не последовательности `rg`, а последовательности `gb`. Тогда на столе остались бы фишки `rrb`. Аналогично, игрок мог бы добиться того, чтобы в конце остались ряды `rrr` или `rbb`. После окончания игры полученные фишки игрок меняет на деньги. Цена фишки зависит от её цвета. Требуется написать программу, определяющую максимальную сумму, которую сможет получить игрок.

### Формат входных данных

В первой строке входных данных содержится число  $K$  ( $1 \leq K \leq 26$ ) - количество цветов фишек. Каждая из следующих  $K$  строк начинается со строчной латинской буквы, обозначающей цвет. Далее в той же строке через пробел следует целое число  $X_i$  ( $1 \leq X_i \leq 150, i = 1..K$ ) - цена фишки соответствующего цвета. В  $(K + 2)$ -ой строке описан ряд фишек, лежащих на столе в начале игры. Ряд задается  $L$  строчными латинскими буквами ( $1 \leq L \leq 150$ ), которые обозначают цвета фишек ряда. В следующей строке содержится число  $N$  ( $1 \leq N \leq 150$ ) - количество последовательностей, которые были объявлены крупье. В следующих  $N$  строках записаны эти последовательности. Гарантируется, что сумма длин этих  $N$  строк не превосходит 150 символов, и все они непустые.

### Формат выходных данных

Выведите единственное целое число — максимальную сумму денег, которую может получить игрок.

### Пример

стандартный ввод	стандартный вывод
3 v 3 l 1 u 2 luvu 3 luv vul uuu	6

## Задача Е. Покраска забора

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.7 секунд
Ограничение по памяти:	256 мегабайт

256 мегабайт

Мэр города Многоярославца решил построить перед своим домом забор из  $n$  деревянных досок и нанять лучшего маляра города для его покраски. Поскольку забор должен стать главной достопримечательностью города, лучший дизайнер города для каждой доски назначил тщательно выбранный цвет, в который она должна быть покрашена.

Для покраски главный маляр решил применить новейшую технологию, специально разработанную им для выполнения этого задания. Покраской забора будет заниматься специальный робот, который за один час может покрасить произвольный отрезок забора (набор соседних досок) в некоторый цвет. Поскольку задание должно быть выполнено как можно быстрее, требуется составить программу для робота, которая позволит достичь требуемой раскраски за минимальное время. Оставить какую-то из досок непокрашенной, естественно, запрещается.

### Формат входных данных

В первой строке входного файла записано число  $n$  ( $1 \leq n \leq 300$ ), где  $n$  количество досок в заборе. Вторая строка содержит строку из  $n$  символов, описывающую требуемую покраску забора. Цвета обозначаются заглавными латинскими буквами.

### Формат выходных данных

В первой строке выходного файла выведите  $m$  - наименьшее возможное время покраски забора в часах. Следующие  $m$  строк должны содержать программу покраски для робота. Каждая строка должна содержать два числа  $l_i$  и  $r_i$ , а также заглавную букву латинского алфавита, задающую цвет  $c_i$  и означает, что робот должен покрасить участок забора с  $l_i$  по  $r_i$ - доску в цвет  $c_i$  (если длина забора  $n$ , должно выполняться  $1 \leq l_i \leq r_i \leq n$ ).

### Примеры

стандартный ввод	стандартный вывод
5 АВВСА	3 1 5 А 2 3 В 4 4 С
2 АА	1 1 2 А

## Задача F. Вупсень и Пупсень - не комфортная

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 3.5 секунд  
Ограничение по памяти: 128 мегабайт

Вупсень очень любит давать задачи на поиск наибольшей общей подпоследовательности. Пупсень очень любит давать задачи на поиск наибольшей правильной скобочной подпоследовательности. Нет ничего удивительного в том, что они решили объединиться и подготовить очень сложную задачу на поиск наибольшей общей правильной скобочной подпоследовательности.

Подпоследовательностью строки  $a$  называется такая строка  $b$ , которую можно получить удалением из строки  $a$  символов на каких-либо (возможно, никаких) позициях.

Последовательность круглых скобок называется *правильной* в следующих случаях:

1. Если она пустая.
2. Если она состоит из правильной скобочной последовательности, заключённой в скобки.
3. Если она состоит из двух правильных скобочных последовательностей, записанных одна за другой.

Вам даны две строки  $s$  и  $t$ , состоящие из круглых открывающих и закрывающих скобок. Найдите правильную скобочную последовательность  $w$  максимальной длины, являющуюся подпоследовательностью строк  $s$  и  $t$ .

### Формат входных данных

Две строки  $s$  и  $t$  из круглых скобок, длины которых не превосходят  $n$  ( $1 \leq n \leq 700$ ), по одной в строке. Любая из строк (в том числе обе) может быть пустой.

### Формат выходных данных

Выведите одну строку  $w$  — наибольшую общую правильную скобочную подпоследовательность исходных строк  $s$  и  $t$ . Если таких строк несколько, разрешается вывести любую из них.

### Примеры

стандартный ввод	стандартный вывод
<pre>()()()() )()()</pre>	<pre>((())()</pre>
<pre>))( (()</pre>	

## Задача G. AliKingspress

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Для экстренного пополнения боеприпасов и вооружения агенты «Кингсман» пользуются службой «AliKingspress». Помимо запроса помощи с вооружением, можно также делать другие запросы, однако уже не бесплатно, а за бонусные баллы. Баллы можно получать каждый день, заходя в специальное приложение. В первый день пользователь получает  $a_1$  баллов, во второй —  $a_2$  баллов, ..., в  $n$ -й день —  $a_n$  баллов. После этого, заходя каждый день, пользователь будет все еще получать  $a_n$  бонусов. Если же пропустить один или несколько дней и не заходить в приложение, при следующем заходе начисление бонусов опять начнется с  $a_1$ .

Эггси посчитал, что для выполнения всех дополнительных запросов, которые он хочет, нужно  $x$  бонусов. Так как он перфекционист, лишние бонусы ему не нужны, он хочет накопить их ровно  $x$ , ни больше, ни меньше. Однако сделать это нужно как можно быстрее, потому что долго ждать он не намерен. Задачу нахождения минимального количества дней, требуемого для этого, он поручил вам — своего верному программисту, пока он сам спасает мир. Помогите ему!

### Формат входных данных

В первой строке содержится два числа  $n$  и  $x$  — количество различных бонусов, а также суммарное количество бонусов, которое нужно набрать Эггси ( $1 \leq n \leq 100, 1 \leq x \leq 10^6$ ).

Во второй строке содержится  $n$  чисел  $a_i$  — размеры бонусов в зависимости от количества дней захода в приложение ( $1 \leq a_i \leq 1000$ ).

### Формат выходных данных

В единственной строке выведите минимальное количество дней, нужное для получения ровно  $x$  бонусов или  $-1$ , если набрать ровно  $x$  бонусов невозможно.

### Примеры

стандартный ввод	стандартный вывод
4 14 1 2 3 4	5
5 12 1 4 2 6 3	6
3 8 3 4 2	-1

### Замечание

В первом примере Эггси может заходить в приложение 5 дней подряд и получить, соответственно,  $1 + 2 + 3 + 4 + 4 = 14$  бонусов.

Во втором тестовом примере Эггси может зайти в приложение три дня подряд, затем пропустить один день, а затем зайти еще два дня подряд. В результате он получит  $1 + 4 + 2 + 1 + 4 = 12$  бонусов и потратит на это  $3 + 1 + 2 = 6$  дней.

## Задача Н. Морти и подпоследовательности

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

Все знают, как Рик и Морти любят путешествовать и влезать в авантюры! И новое путешествие не исключение! Перед тем как отправиться, Рик попросил Морти помочь ему справиться с одной жизненно-важной задачей, без которой путешествию не состояться. Маленький Морти уже попытался справиться, но у него ничего не вышло, именно поэтому он решил обратиться за помощью к вам!

Задача, которую дал ему Рик звучит следующим образом: дан массив  $a$  из  $n$  целых положительных чисел. Для всех целых  $k$ , для которых выполняется неравенство  $1 \leq k \leq n$  нужно определить, сколько какое максимальное число элементов можно оставить, убрав некоторые, так, чтобы оставшийся массив можно было разбить на подотрезки, каждый из которых — возрастающая последовательность, длины не меньше  $k$ .

Последовательность  $a_{i_1}, a_{i_2}, \dots, a_{i_p}$  называется возрастающей подпоследовательностью в массиве  $a$ , если  $a_{i_1} < a_{i_2} < \dots < a_{i_p}$ .

Размер последовательности — количество элементов, которые принадлежат последовательности.

### Формат входных данных

Первая строка входных данных содержит одно целое число  $n$  ( $1 \leq n \leq 300$ ) отвечающее за длину массива. На второй строке содержится массив  $a$  из  $n$  целых чисел,  $1 \leq a_i \leq 10^9$ .

### Формат выходных данных

В единственной строке выведите  $n$  чисел  $b_i$  — максимальное число элементов, которые войдут в непересекающиеся возрастающие подотрезки размера не менее  $i$  путем исключения некоторого (возможно нулевого) числа элементов из исходного массива.

### Примеры

стандартный ввод	стандартный вывод
3 1 2 3	3 3 3
2 1 1	2 0
5 1 4 3 2 9	5 4 3 0 0

### Замечание

Рассмотри третий пример. Для  $k = 1$ , ответ равен 5, так как каждый элемент по отдельности является возрастающей последовательностью. Для  $k = 2$  максимальный ответ достигается путем избавления, например, от числа 4, разбивая оставшийся массив на два отрезка длины 2, которые являются возрастающими последовательностями. Для  $k = 3$  максимальный ответ можно достичь удалив элементы со значениями 4 и 3, в результате получив один отрезок, который является возрастающей последовательностью длины 3.



## Задача I. Подозрительная строка

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

Расследуя убийство, Эркюль Пуаро нашёл подозрительную строку из маленьких латинских букв. Он предположил, что эта строка принадлежит доктору Арбэтноу, и хочет проверить свою догадку.

Доктору Арбэтноу очень нравятся палиндромы, поэтому Пуаро хочет определить, похожа ли найденная им строка на палиндром. Пуаро считает строку похожей на палиндром, если можно переставить в ней буквы так, чтобы она стала палиндромом и при этом каждая буква переместилась не более чем на одну позицию.

Помогите великому сыщику — определите, похожа ли найденная им строка на палиндром.

### Формат входных данных

Во входном файле задана строка из маленьких латинских букв длины не более  $10^5$ .

### Формат выходных данных

Выведите YES, если строка похожа на палиндром, или NO в противном случае.

### Примеры

стандартный ввод	стандартный вывод
baaacba	YES
steponnopets	YES
noooo	NO

### Замечание

В первом тесте из строки можно получить палиндром abacaba.

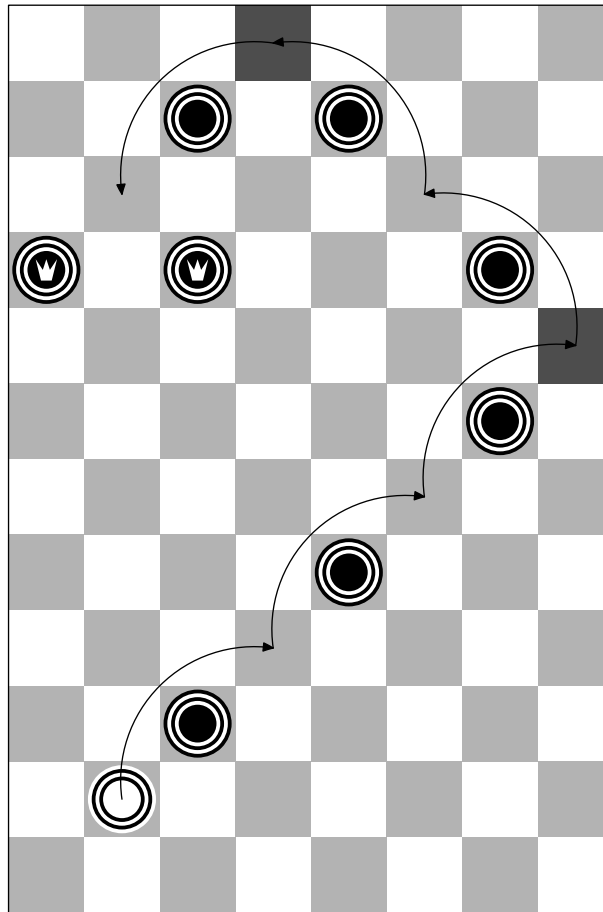
Во втором тесте строка и так является палиндромом.

## Задача J. Достойный финал

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Халк и Танос, решив, что на сегодня довольно драк, решили сыграть в шашки, чтобы выяснить, кто круче. Однако играют они по особым правилам.

- Поле имеет размеры  $h \times w$ , его строки пронумерованы снизу вверх числами от 1 до  $h$ , а столбцы — слева направо числами от 1 до  $w$ .
- Клетки, у которых номера столбца и строки имеют одинаковую четность, покрашены в черный цвет, а остальные — в белый.
- В некоторых черных клетках находятся белые или черные шашки. Ни в какой клетке не находится более одной шашки.
- Игроки ходят по очереди, за ход можно либо ничего не сделать, либо походить шашкой своего цвета: Халк — белой, Танос — черной.
- Если игрок ходит шашкой, он совершает ею последовательность *взятий*. А именно, пусть шашка игрока находится в клетке  $A$ , в соседней по углу клетке  $B$  находится шашка противника, а следующая в том же направлении клетка  $C$  пустая. Тогда игрок может *взять* шашкой в клетке  $A$  шашку в клетке  $B$ : при этом шашка игрока перемещается с  $A$  на  $C$ , а «перепрыгнутая» шашка противника снимается с поля, то есть клетка  $B$  оказывается пустой. Если после взятия походившая шашка способна взять ещё какую-нибудь шашку противника, то игрок может продолжать совершать *взятия* своей шашкой, пока желает.
- В игре есть дополнительное правило: нельзя за один ход более двух раз менять направление, в котором шашка движется, совершая взятия. Таким образом, в приведенной на рисунке ниже ситуации белая шашка не могла за свой ход взять ни одну из двух шашек, в которых нарисована корона: чтобы взять левую из них, ей необходимо было бы выпрыгнуть из поля, а чтобы взять правую — в третий раз сменить направление прыжков (первые два изменения направления произошли в выделенных клетках).



После нескольких ходов у Халка осталась всего одна шашка, однако сдаваться он не собирается. Помогите ему за один ход взять как можно больше шашек Таноса!

### Формат входных данных

В первой строке находится три целых числа  $h$ ,  $w$ ,  $n$  — высота, ширина поля и количество черных шашек ( $1 \leq h, w, n \leq 250\,000$ ).

Во второй строке находится два целых числа  $p$ ,  $q$  — номера строки и столбца, на пересечении которых находится белая шашка ( $1 \leq p \leq h$ ,  $1 \leq q \leq w$ ).

В каждой из следующих  $n$  строк находится по два целых числа  $r_i$ ,  $s_i$  — номера строки и столбца, на пересечении которых находится  $i$ -я черная шашка ( $1 \leq r_i \leq h$ ,  $1 \leq s_i \leq w$ ).

Гарантируется, что ни у какой черной шашки пара координат не совпадает с парой координат другой черной или белой шашки. Гарантируется, что все шашки находятся в клетках черного цвета.

### Формат выходных данных

Выведите одно целое число — наибольшее количество черных шашек, которое за один ход может взять белой шашкой Халк, следуя указанным выше правилам.

## Пример

стандартный ввод	стандартный вывод
12 8 8	6
2 2	
9 1	
9 3	
11 3	
3 3	
5 5	
11 5	
9 7	
7 7	

## Задача К. Трудности переписки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Этим летом Джек ездил в летнюю школу в России. Там он завел много новых друзей, а также встретил красивую девушку. По возвращении домой родители подарили Джеку новый ноутбук, и теперь он всегда может быть на связи со своими новыми друзьями. Естественно, получив подарок, Джек сразу стал переписываться со своей подругой Ирой.

Отправив несколько сообщений, Джек заметил, что ноутбук, а точнее его клавиатура, работает не так, как он ожидал. В процессе ввода сообщения у ноутбука иногда внезапно срабатывает клавиша «Home», в результате чего курсор ввода перемещается в начало строки. Так, например, если у Джека в процессе ввода строки «irailikeyou» клавиша «Home» сработала после ввода букв «а» и «у», то получится строка «ouilikeyira». Джек планировал набрать строку  $s$ , нажимая по очереди на соответствующие клавиши. Закончив набор, он посмотрел на экран и увидел строку  $t$ . Теперь он хочет понять, может ли она быть результатом его ввода, если единственная неисправность его ноутбука — лишние срабатывания клавиши «Home», либо у его ноутбука есть еще проблемы. Помогите Джеку.

### Формат входных данных

В первой строке задано число  $n$  — длина строк  $s$  и  $t$  ( $1 \leq n \leq 5000$ ). Во второй строке задана последовательность маленьких латинских букв длины  $n$  — строка  $s$ . В третьей строке задана последовательность маленьких латинских букв длины  $n$  — строка  $t$ .

### Формат выходных данных

Выведите «YES», если из строки  $s$  могла получиться строка  $t$ , иначе выведите «NO».

### Пример

стандартный ввод	стандартный вывод
11 irailikeyou ouilikeyira	YES

## Задача L. Кулинарное шоу

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

Уолтер *обожает* смотреть кулинарное шоу по вечерам. К сожалению, в ближайшее время он будет очень занят, поэтому решил написать программу, которая будет скачивать выпуски шоу для него.

Выпуски будут выходить по одному в день. Каждый выпуск — это отдельный файл, размер файла, содержащего  $i$ -й выпуск, равен  $s_i$  байт.

Программа Уолтера будет действовать следующим образом. Она заранее выбирает набор размеров блоков  $x_j$  — эта последовательность будет одинаковой каждый день. Каждый день она один за другим будет отправлять на сервер запросы, где  $j$ -й запрос представляет собой «загрузить очередные  $x_j$  байт». В ответ на такой запрос сервер возвращает пакет данных, содержащий очередные  $x_j$  байт файла, а также заголовок, содержащий  $k$  байт различной служебной информации. Таким образом, размер пакета равен  $x_j + k$  байт, при этом значение  $k$  одно и то же для всех запросов.

Когда в результате некоторого запроса скачивается последний байт файла, программа завершает свою работу и не делает дальнейших запросов к серверу. Однако протокол устроен таким образом, что размер пакета равен  $x_j + k$ , даже если был достигнут конец файла и в действительности было загружено меньше  $x_j$  байт полезной информации.

Интернет у Уолтера дорогой, поэтому он хочет, чтобы суммарный размер всех загруженных пакетов был как можно меньше.

Уолтер заранее знает размер каждого выпуска.

Уолтер неплохо варит макароны, но ничего не знает о программировании, поэтому ему нужна ваша помощь. Помогите ему узнать, какой минимальный суммарный размер пакетов придётся скачать, чтобы скачать все выпуски шоу.

### Формат входных данных

В первой строке входного файла заданы целые числа  $n$  и  $k$  — количество выпусков и размер заголовка пакета ( $1 \leq n \leq 10000$ ;  $0 \leq k \leq 10^9$ ).

Во второй строке задано  $n$  целых чисел  $s_i$  — размеры выпусков ( $1 \leq s_i \leq 10^9$ ).

### Формат выходных данных

Выведите единственное число — ответ на задачу

### Примеры

стандартный ввод	стандартный вывод
4 1000 100 200 200 800	6400
4 0 100 200 800 200	1300

### Замечание

В первом примере можно сначала загружать 200 байт, а затем 600. Во втором примере можно загружать по 100 байт за запрос.