Задача A. Циклическое расстояние Хэмминга

Имя входного файла:	`hamming.in`
Имя выходного файла:	`hamming.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Расстояние Хэмминга $h(S, T)$ между двумя строками одинаковой длины $S_1 S_2 \ldots S_n$ и $T_1 T_2 \ldots T_n$ — это количество позиций i $(1 \leqslant i \leqslant n)$ таких, что $S_i \neq T_i$. Например, $h(\texttt{abaca}, \texttt{addch}) = 3$, поскольку символы в позициях 1 и 4 в этих двух строках совпадают, а в позициях 2, 3 и 5 соответствующие символы различны.

Циклическая перестановка строки $S = S_1 S_2 \ldots S_n$ — это строка $S_k S_{k+1} \ldots S_n S_1 S_2 \ldots S_{k-1}$ для некоторого k; в частности, для $k = n$ циклическая перестановка совпадает с самой строкой.

Назовём *циклическим расстоянием Хэмминга* $c(S, T)$ наименьшее из расстояний Хэмминга между некоторыми двумя циклическими перестановками S и T. Например, для строк $S = \texttt{abaca}$ и $T = \texttt{addch}$ расстояние $c(S, T) = 3$, а для $S = \texttt{aba}$ и $T = \texttt{bab}$ $h(S, T) = 3$, но $c(S, T) = 1$.

По двум данным строкам одинаковой длины найдите циклическое расстояние Хэмминга между ними.

Формат входного файла

В первых двух строкчах входного файла заданы строки S и T одинаковой длины, по одной на строчку. Длина заданных строк — от 1 до 1000 символов; строки состоят только из маленьких букв `a-z` латинского алфавита.

Формат выходного файла

Выведите в выходной файл одно число — $c(S, T)$.

Примеры

hamming.in	hamming.out
abaca addch	3
aba bab	1
tmxyztmxyztm xyztmtmxyztm	0
a b	1

Задача B. Лишнее число

Имя входного файла:	`excess.in`
Имя выходного файла:	`excess.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В штаб секретной службы поступило сообщение от одного из агентов. Поступившее сообщение в зашифрованном виде представляет собой последовательность чисел, и лишь специальная программа способна расшифровать его и получить связный текст.

Обычно программа-расшифровщик быстро и бесшумно выдаёт связистам расшифрованный текст, но в этот раз вместо текста от программы поступил сигнал тревоги, свидетельствующий о том, что при пересылке сообщение было взломано или просто повреждено.

Корректное зашифрованное сообщение — это последовательность из $4 \cdot k$ целых чисел, в котором k различных чисел присутствуют по 4 раза каждое; для расшифровки даже не важны значения этих чисел, а важен лишь их порядок.

Однако, изучив зашифрованное сообщение, связисты обнаружили, что в нём $4 \cdot k + 1$ число. При этом ровно одно число является «лишним», то есть при его удалении зашифрованное сообщение становится корректным сообщением из $4 \cdot k$ чисел (возможно, четыре из них равны удалённому числу).

Связисты решили, что на будущее им нужна программа, которая находит такое «лишнее» число автоматически. Помогите им написать такую программу.

Формат входного файла

В первой строке входного файла задано число $N = 4 \cdot k + 1$, где N и k целые, и $1 \leqslant k \leqslant 10\,000$. В последующих N строках находятся числа $A_1, A_2, \ldots, A_N$, по одному числу в каждой — зашифрованное сообщение. Известно, что $0 \leqslant A_i \leqslant 1\,000\,000$.

Формат выходного файла

В первую строку выходного файла выведите «лишнее» число из набора A_i.

Примеры

excess.in	excess.out
5 4 1 4 4 4	1
9 1 3 3 1 3 3 3 1 1	3

Задача C. Диаметр

Имя входного файла:	`diameter.in`
Имя выходного файла:	`diameter.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Диаметром множества называется максимальное из расстояний между двумя его точками. Так, диаметр окружности — это её диаметр в обычном смысле, диаметром прямоугольника является расстояние между противоположными вершинами.

В этой задаче предлагается найти диаметр выпуклого многоугольника, вписанного в окружность радиуса 1.

Формат входного файла

В первой строке входного файла задано целое число N ($3 \leqslant N \leqslant 10\,000$). В последующих N строках даны углы $A_1, A_2, \ldots, A_N$ в градусах, по одному на строке — полярные углы вершин многоугольника в порядке следования против часовой стрелки. $0 \leqslant A_i < 360$; все A_i даны не более чем с тремя знаками после десятичной точки. Никакие два угла не совпадают.

Формат выходного файла

Выведите в выходной файл одно число — диаметр данного многоугольника. Число должно иметь не менее шести точных знаков после десятичной точки.

Примеры

diameter.in	diameter.out
3 0 60 90	1.4142135624
4 0 90 180 270	2
5 330 345 0.001 15 30	1.000000