TASK	TRENER	KUŠAČ	RATAR	LOPOV	ORGANIZATOR	SLASTIČAR
source code	trener.pas trener.c trener.cpp	kusac.pas kusac.c kusac.cpp	ratar.pas ratar.c ratar.cpp	lopov.pas lopov.c lopov.cpp	organizator.pas organizator.c organizator.cpp	slasticar.pas slasticar.c slasticar.cpp
input	standard input (stdin)					
output	standard output (stdout)					
time limit	1 second	1 second	1 second	1 second	1 second	3 seconds
memory limit	32 MB	32 MB	32 MB	32 MB	32 MB	64 MB
point value	50	80	100	120	140	160
	650					

Problems translated from Croatian by: **Ivan Pilat**

Mirko has been moving up in the world of basketball, starting as a mere spectator, mastering snack salesmanship, finally reach the coveted position of the national team coach. He is now facing a difficult task: selecting the five primary players for the upcoming match against Tajikistan.

Since Mirko is incredibly lazy, he doesn't bother remembering players' names, let alone their actual skills. That's why he has settled on selecting five players who share the same first letter of their surnames, so that he can remember them more easily. If there are no five players sharing the first letter of their surnames, Mirko will simply forfeit the game!

In order to obtain insight into possibilities for his team, Mirko wants to know all the different letters that his primary team's surnames may begin with.

INPUT

The first line of input contains the positive integer **N** ($1 \leq \mathbf{N} \leq 150$), the number of players that Mirko has available.

Each of the following **N** lines contains one word (at most 30 characters long, consisting only of lowercase English letters), a surname of one of the players.

OUTPUT

If there are no five players that Mirko can select matching his criteria, output a single line containing the word "PREDAJA"[1] (without quotes). Otherwise, output all possible first letters of representation player surnames, sorted lexicographically, in a single line with no spaces.

SAMPLE TESTS

input	**input**
18 babic keksic boric bukic sarmic balic kruzic hrenovkic beslic boksic krafnic pecivic klavirkovic kukumaric sunkic kolacic kovacic prijestolonasljednikovic	6 michael jordan lebron james kobe bryant

1 Croatian word for surrender

output	**output**
bk	PREDAJA

Clarification of the first example: Mirko can choose between teams with all players' surnames begining with either 'k' or 'b'.

Mirko has given up on the difficult coach job and switched to food tasting instead. Having skipped breakfast like a professional connoisseur, he is visiting a Croatian cured meat festival. The most renowned cook at the festival, Marijan Bajs, has prepared **N** equal sausages which need to be distributed to **M** tasters such that each taster gets a precisely equal amount. He will use his trusted knife to cut them into pieces.

In order to elegantly divide the sausages, the **number of cuts** splitting individual sausages must be **as small as possible**. For instance, if there are two sausages and six tasters (the first test case below), it is sufficient to split each sausage into three equal parts, making a total of four cuts. On the other hand, if there are three sausages and four tasters (the second test case below), one possibility is cutting off three quarters of each sausage. Those larger parts will each go to one of the tasrers, while the fourth taster will get the three smaller pieces (quarters) left over.

Mirko wants to try the famous sausages, so he volunteered to help Bajs. Help them calculate the minimum total number of cuts needed to carry out the desired division.

INPUT

The first and only line of input contains two positive integers, **N** and **M** ($1 \leq \mathbf{N}, \mathbf{M} \leq 100$), the number of sausages and tasters, respectively.

OUTPUT

The first and only line of output must contain the required minimum number of cuts.

SAMPLE TESTS

input	**input**	**input**
2 6	3 4	6 2
output	**output**	**output**
4	3	0

After Mirko's failed stint as a coach and a passing obsession with Croatian meat delicacies, his weight problems have motivated him to work hard as a farmer. He has moved to a village where his friend Slavko lives. Farmers in the village share a large common plot of land in the shape of a **N×N** square, divided into $\mathbf{N}^2$ unit squares. A unit square at coordinates[2] **(i, j)** brings in the income of $\mathbf{A_{ij}}$, which can be negative (for example, if the square has to be maintained but is not cultivated). The farmers always divide the common land into smaller **rectangular fields** with edges **parallel** to the common land edges.

Slavko is skeptical of Mirko since his failure as a coach, so he insists that both of them are assigned land with the **same total income**, but also thet the two plots share **exactly one** common corner so that the two friends can keep an eye on each other (Slavko knows that Mirko is prone to mischief). The common corner must be the only point where the two plots meet, in order to prevent border-related arguments.

You are given a description of the common land plot. Find the total number of plot pairs that satisfy Slavko's criteria.

INPUT

The first line of input contains the positive integer **N** ($1 \leq \mathbf{N} \leq 50$), the dimensions of the common land plot.

Each of the following **N** lines contains **N** space-separated numbers $\mathbf{A_{ij}}$ ($-1000 < \mathbf{A_{ij}} < 1000$), the income provided by the respective cell.

OUTPUT

The first and only line of output must contain the totl number of plot pairs satisfying the given condition.

SCORING

In test data worth at least 40% of total points, **N** will be at most 10.

SAMPLE TESTS

input	**input**	**input**
3 1 2 3 2 3 4 3 4 8	4 -1 -1 -1 -1 1 2 3 4 1 2 3 4 1 2 3 4	5 -1 -1 -1 -1 -1 -2 -2 -2 -2 -2 -3 -3 -3 -3 -3 -4 -4 -4 -4 -4 -5 -5 -5 -5 -5
output	**output**	**output**
7	10	36

2 The first number in the parenthesis denotes the row, and the second number denotes the column

Clarification of the first example: The possible rectangle pairs are:
(0,0)-(1,1) and (2,2)-(2,2), (1,0)-(1,0) and (0,1)-(0,1), (2,0)-(2,0) and (1,1)-(1,1), (1,1)-(1,1) and (0,2)-(0,2), (2,1)-(2,1) and (1,2)-(1,2), (2,0)-(2,1) and (0,2)-(1,2), (1,0)-(2,0) and (0,1)-(2,0).

The difficult economic situation in the country and reductions in government agricultural subsidy funding have caused Mirko to change his career again, this time to a thief. His first professional endeavour is a jewellery store heist.

The store contains **N** pieces of jewellery, and each piece has some mass $\mathbf{M_i}$ and value $\mathbf{V_i}$. Mirko has **K** bags to store his loot, and each bag can hold some maximum mass $\mathbf{C_i}$. He plans to store all his loot in these bags, but **at most one** jewellery piece in each bag, in order to reduce the likelihood of damage during the escape.

Find the maximum total jewellery value that Mirko can "liberate".

INPUT

The first line of input contains two numbers, **N** and **K** ($1 \leq \mathbf{N}, \mathbf{K} \leq 300\,000$).

Each of the following **N** lines contains a pair of numbers, $\mathbf{M_i}$ and $\mathbf{V_i}$ ($1 \leq \mathbf{M_i}, \mathbf{V_i} \leq 1\,000\,000$).

Each of the following **K** lines contains a number, $\mathbf{C_i}$ ($1 \leq \mathbf{C_i} \leq 100\,000\,000$).

All numbers in the input are positive integers.

OUTPUT

The first and only line of output must contain the maximum possible total jewellery value.

SCORING

In test data worth at least 50% of total points, **N** and **K** will be less than 5000.

SAMPLE TESTS

input	**input**
2 1 5 10 100 100 11	3 2 1 65 5 23 2 99 10 2
output	**output**
10	164

Clarification of the second example: Mirko stores the first piece of jewellery into the second bag and the third piece into the first bag.

Unexpected problems with law enforcement have convinced Mirko to take up a less lucrative but less morally ambiguous career: he has become the chief organizer of a team computer science contest.

There are **N** CS clubs that wish to participate in the contest. The presidents of the clubs are quite stubborn and will participate in the contest **only if** the contest team size makes it possible for all club members to participate.

The contest consists of two rounds: **qualifications** and **finals**. All teams that are competing must have an **equal** number of members and **all members** of one team must belong to the same club. Any number of teams from each club can participate in the qualification round, and the **best team** from each club earns a spot in the **finals**.

Mirko is aware that, with a new and unproven contest, he needs publicity. For that reason, he wants to set the team size such that the **number of individual participants** in the **finals** is as **large** as possible. Remember, each club that participates has a right to **one** team in the **finals**. Furthermore, at least **two** clubs must participate in the contest, otherwise the contest would be too boring to attract sponsors.

Determine the maximum possible number of participants in the finals so that Mirko can double check his team size choice.

INPUT

The first line of input contains the positive integer **N** ($2 \leq \mathbf{N} \leq 200\,000$), the number of clubs.

The second line of input contains **N** space-separated integers from the interval [1, 2 000 000], the number of members of each club.

OUTPUT

The first and only line of output must contain the maximum possible number of finalists.

SCORING

In test data worth at least 30% of total points, **N** will be less than 1000.

SAMPLE TESTS

input	**input**	**input**
3 1 2 4	2 1 5	5 4 6 3 8 9
output	**output**	**output**
4	2	9

Clarification of the first example: Mirko decides on 2 members per team, so clubs 2 and 3 participate.

Organizing CS contests didn't prove very lucrative for Mirko, so he has opened an ice cream and pastry shop. The business was flourishing until, one day, the European Union health inspection decided to pay him a visit.

A new directive specifies **M** banned ingredients which cannot appear in food even in trace amounts. Each ingredient has a serial number consisting of digits 0 through 9. The declaration on each food package lists all the serial numbers of ingredients contained in the respective food item.

Mirko must check whether any of his products has a banned ingredient serial number listed on its declaration. However, Mirko, being inept and reckless as always, decided to concatenate all the serial numbers into one long number with length **N** believing that it will make his job easier. He has borrowed a robot from his friend Slavko. The robot is programmed to check whether a serial number A contains another serial number B as a substring. Let us denote the length of B by L. The robot carries out search as follows:

- First, it compares the segment of A from position 1 to position L, digit by digit, with the digits in B. Comparison is stopped when a differing digit is found or when the segments are determined to be equal. If the segments are equal, the search is stopped and the match reported.
- If the segments are not equal, the procedure above is repeated with the segment from 2 to L+1. If those segments aren't equal either, the search continues with segments 3 to L+2, 4 to L+3 etc.
- If the robot doesn't have a sufficient number of digits to obtain a full segment of length L (for example, starting at character 5 in a serial number with length 8, a segment with length 6 is needed), it will pad the number with '#' signs. For example, the segment of "563232" from position 4 to position 10 is "232####".
- If the robot reaches the end of the serial number (having tried out all **N** segments) without having found B, the absence of match is reported.

The robot takes one second for each comparison between two digits, and Slavko charges Mirko one dollar per second for using the robot.

Help Mirko determine how much money he'll have to pay Slavko for pattern matching!

INPUT

The first line of input contains the positive integer **N** ($1 \leq \mathbf{N} \leq 100\,000$), the length of the long serial number.

The second line of input contains **N** digits from 0 to 9, the long serial number.

The third line of input contains the positive integer **M** ($1 \leq \mathbf{M} \leq 50\,000$), the number of banned ingeredients.

Each of the following **M** rows contains a single banned serial number,

A banned serial number will not exceed 100 000 digits in length.

The total length of all banned serial numbers will not exceed 3 000 000 digits.

OUTPUT

Output **M** integers, one per line. Line *i* must contain the dollar amount that Mirko needs to pay Slavko for the search for ingredient serial number *i*.

SCORING

In test data worth at least 20% of total points, the following constraints hold:

- $1 \leq \mathbf{N} \leq 1000$
- $1 \leq \mathbf{M} \leq 500$
- length of a single banned ingredient serial number doesn't exceed 1000

SAMPLE TESTS

input	**input**	**input**
7 1090901 4 87650 0901 109 090	10 5821052680 4 210526 2105 582 105268	3 001 1 11
output	**output**	**output**
7 10 3 4	8 6 3 9	4

Clarification of the first example:
First serial number: the robot finds differing first digits for every segment – a total of 7 comparisons.
Second serial number: tries first position, finding difference immediately (1 comparison). Tries second position, finding difference on the fourth digit (4 comparisons). Tries third position, finding difference immediately (1 comparison).Tries fourth position, finding a match (4 comparisons). Total: 10 comparisons.
Third serial number: finds match immediately (3 comparisons).
Fourth serial number: finds match at second position (1 + 3 = 4 comparisons).
Clarification of the third example:
The robot compares the serial number '11' in order with segments '00' (1 comparison), '01' (1 comparison) and '1#' (2 comparisons). Total: 4 comparisons.