Problem A. Bookmakers

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

This summer programmer Sergey visited the homeland of football. On his first day in England, he discovered that there were more bookmakers near the Wembley Stadium than football fans. The bookmakers were taking bets at a wide range of payout odds. As a programmer, Sergey found it easier to operate with betting coefficients instead of odds, and he easily converted odds into these coefficients: if the odds for some outcome of a match are $m : n$, then the betting coefficient k is $(m + n)/n$. This means that if Sergey placed a bet of x pounds and guessed the outcome correctly, he is paid $k \cdot x$ pounds by the bookmaker (and gets nothing if he was wrong).

Having walked around the stadium, Sergey found out that the best coefficients for the three possible outcomes were 3.5, 3.5, and 3.5 (of course, they were offered by different bookmakers). Sergey placed a bet of 330 pounds on each outcome and was not a bit upset about losing two of the three bets because the third bet brought him 1155 pounds, which was more than he paid to the three bookmakers altogether. In order to cover his travel expenses, Sergey decided to place bets every day distributing them in such a way that he would win as much as possible in the worst case. Write a program that will help Sergey to place bets optimally.

You are given the best coefficients k_1, k_2, and k_3 for the three possible outcomes of a match. Find the maximal sum of money that Sergey will receive in the worst case if the total amount of the bets he places is 1000 pounds.

Input

The single line of the input contains the numbers k_1, k_2, and k_3 separated by a space. The numbers are in the range from 0.01 to 100.

Output

In the single line of the output, give the maximal amount of money that Sergey gets back in the worst case. You may assume that bets and payouts can be fractional numbers. Round the answer to an integer according to the standard mathematical rule.

Examples

input.txt	output.txt
3.5 3.5 3.5	1167
1.25 10.0 5.75	931

Problem B. Cheese

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

Plus il y a de fromage, plus il y a de trous
or plus il y a de trous, moins il y a de fromage
donc plus il y a de fromage, moins il y a de fromage
Paradoxe de l'emmental

Programmer Sasha went downhill skiing in the Swiss Alps this summer. Switzerland is famous for its banks and cheese. Sasha was not interested in banks, but he visited a cheese factory. He learned there that Swiss people treated cheese even more seriously than money. For example, they put cheese into vacuum packs in pieces of exactly 500 grams. If you think that it is very easy to cut off such pieces, then you are mistaken: there are holes in Swiss cheese, which must be taken into account. Fortunately, modern technologies make it possible to determine exactly the size and location of all cavities in a piece of cheese.

Cheese is fed to a cutting machine as a long bar of square section 10×10 cm and length 1 m. Assume that all cavities are spherical and do not intersect each other and the borders of the bar. The machine can cut the bar at right angle to the long edges using a micrometer scale (that is, a grid of size 1 μm). A special computer determines the value the coordinate along the long edge of the bar at which a cut should be made so that the weight of the next piece be exactly 500 grams. This value is rounded to micrometers and the machine cuts the bar. Then the computer determines where the next cut should be made, and so on, until the weight of the remaining piece is less than 500 grams. If it turns out that the last cut should be made at the coordinate of exactly 1 meter, then, of course, this cut is not made.

Write a program that determines the coordinates of the cuts exactly as the Swiss computer does.

Input

The first line contains the number n of cavities in a bar of cheese ($0 \le n \le 100$). The next n lines describe these cavities in the format $x_i\ y_i\ z_i\ r_i$, where (x_i, y_i, z_i) are the coordinates of the center of a cavity and r_i is its radius. The size of the bar is $10 \times 100 \times 10$ along the x, y, and z axes, respectively. The coordinate origin is one of the corners of the bar; the unit of measure is 1 cm. It is known that 1 cm^3 of Swiss cheese weighs 1 gram.

Output

Output the coordinates of the cuts on the scale of the cutting machine. In the first line output the number of cuts, and in the following lines give the coordinates of the cuts in micrometers measured from the beginning of the bar. Separate numbers with spaces and/or line feeds.

Example

input.txt	output.txt
2 3.2 37.2 1.8 1 4.2 66.6 5.5 2.5	19 50000 100000 150000 200000 250000 300000 350000 400419 450419 500419 550419 600419 651114 706964 756964 806964 856964 906964 956964

Problem C. Pharaohs' Secrets

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

When programmer Alex was in Egypt, he not only swam in the Red Sea and went sightseeing, but also studied history. When Alex visited the place where an archeological dig of an ancient temple was carried out, an excavation worker complained to him that they had to drag very heavy statues from place to place every day. This was because some Egyptologist had read in an ancient papyrus that if the statues were arranged in a special order, then some ancient hiding-place would open. When the temple had been dug out, these statues had stood as soldiers, forming a rectangle. Some statues were identical, so there were several types of statues. They were to be arranged into a rectangle of the same dimensions on the same place with all rows and columns symmetric with respect to their middles. This meant that the statues standing in the same row or column at equal distances to its ends had to be of the same type.

Alex offered his help. He wants to find the way to transform the rectangle into a symmetric one by means of the minimal number of moves.

Input

The first line contains the dimensions of the rectangle n and m ($2 \le n, m \le 20$). These integers are even. Each of the next n lines contains m lowercase English letters. Each letter denotes the type of the statue that stands in the rectangle at this position.

Output

Output the minimal number of statues that should be moved in order to make a symmetric rectangle. It is guaranteed that this is possible.

Example

input.txt	output.txt
4 4 abxa xyyb xyyx abba	2

The arrangement in the example can be transformed to a symmetric one in only two moves: first the statue of the type "x" from the upper row should be moved to the place in the rightmost column where there is the statue of the type "b", and this statue then should moved to the place where the first statue stood. After all moves each place must be occupied by exactly one statue, but during the moving process there can be several statues at the same place.

Problem D. Penguins

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

Programmer Denis has been dreaming of visiting Antarctica since his childhood. However, there are no regular flights to Antarctica from his city. That is why Denis has been studying the continent for the whole summer using a local cinema. Now he knows that there are several kinds of penguins:

- Emperor Penguins, which are fond of singing;
- Little Penguins, which enjoy dancing;
- Macaroni Penguins, which like to go surfing.

Unfortunately, it was not said in the cartoons which kind of penguins is largest in number. Denis decided to clarify this. He watched the cartoons once more and every time he saw a penguin he jotted down its kind in his notebook. Then he gave his notebook to you and asked you to determine the most numerous kind of penguins.

Input

The first line contains the number n of entries in the notebook ($1 \le n \le 1000$). In each of the next n lines, there is the name of a kind of penguins, which is one of the following: "Emperor Penguin," "Little Penguin," and "Macaroni Penguin."

Output

Output the most numerous kind of penguins. It is guaranteed that there is only one such kind.

Example

input.txt	output.txt
7 Emperor Penguin Macaroni Penguin Little Penguin Emperor Penguin Macaroni Penguin Macaroni Penguin Little Penguin	Macaroni Penguin

Problem E. Threeprime Numbers

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

Rest at the sea is wonderful! However, programmer Pasha became awfully bored of lying on a beach in Turkey; so bored that he decided to count the quantity of three-digit prime numbers. This turned out to be so interesting that he then started to study threeprime numbers. Pasha calls an integer a threeprime number if any three consecutive digits of this integer form a three-digit prime number. Pasha had already started working on the theory of the divine origin of such numbers when some vandals poured water on Pasha and cried some incomprehensible words like "Sonnenstich!", "Colpo di sole!", and "Coup de soleil!"

You are to continue Pasha's work and find out how often (or rare) threeprime numbers are.

Input

The input contains an integer n ($3 \le n \le 10000$).

Output

Output the quantity of n-digit threeprime numbers calculated modulo $10^9 + 9$.

Example

input.txt	output.txt
4	204

Problem F. Flying Pig

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

This summer programmer Eugene visited Australia. It turned out that in this wonderful country one could not only watch kangaroos and emus, visit the world-famous opera house, and swim in the warm sea, but also enjoy the unique "Flying Pig Show," which took place not far from Sidney. No wonder the show attracts crowds of people: flying pigs with visible pleasure flop into a pool. Before plunging into the water, they take a run and jump from a platform fixed at a height of almost four meters above the water level. In order to gape at the amazing pigs, the crowd gathers about an hour before the start of the show.

After the performance, the organizers choose the most popular pig and present a symbolic prize to it. The popularity of a pig is calculated on the basis of the audience's likings: each guest gives a mark (an integer not exceeding 3 in absolute value) to each pig. Each pig's marks are written as a sequence and then a nonempty subsequence of consecutive numbers is chosen in which the product of numbers is maximal. This product is taken by the organizers as the popularity of the pig.

The show became so popular that the number of guests is very large. That is why it is not always easy to choose the best pig. Eugene offered the organizers his help in automating the process of calculating the popularities of pigs.

Input

The first line contains the number of guests n at the "Flying Pig Show" ($1 \le n \le 50000$). The next line contains n integers, which are the guests' marks for a pig, in the order the organizers write them down.

Output

Output the popularity of the pig.

Example

input.txt	output.txt
3 1 2 -1	2

Problem G. Jamaica

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

Programmer Andrey is very lucky. The Governor-General of Jamaica invited him to visit this wonderful island. The whole trip, including plenty of entertainment, will be completely free for Andrey. He will only have to help the Jamaican Ministry of Transport and write a small program.

The point is that the Jamaican government decided to construct a new network of expressways, in order to boost the economic growth of the country. Each two cities are to be connected by a road going along a straight line. One road may connect several cities if they lie on the same straight line. Jamaican economists are sure that the new network will minimize transportation expenses. In order to estimate the cost of the project, it is required to determine the total length of the roads to be constructed, and this is the job for Andrey.

Input

The first line contains the number n of cities in Jamaica ($1 \le n \le 300$). The next n lines contain the coordinates of the cities. In each of these lines, there are two integers x_i and y_i separated by a space ($0 \le x_i, y_i \le 10000$). There are no cities with coinciding coordinates.

Output

Output the total length of the roads rounded to the nearest integer.

Example

input.txt	output.txt
4 0 0 0 100 100 0 50 50	412

Problem H. Sokoban

Input file:	input.txt
Output file:	output.txt
Time limit:	5 seconds
Memory limit:	64 megabytes

For the period of his vacation, programmer Stas found a job with the Japanese computer company *Thinking Rabbit*. At first glance, the idea seemed marvelous: he would go abroad, earn some money, and learn from his Japanese colleagues. However, it turned out that the company did not want programmers without good knowledge of Japanese. Therefore, Stas was sent to work as a storekeeper (in Japanese, this profession was called *soko-ban*).

Stas had to put the storehouse to order. Every morning he was given a sheet of paper with a scheme of the room in the storehouse where he had to work that day. The scheme showed the places where he had to put containers. For some reason, the management of the company did not bother about which container would be put to which place; they only wanted all containers to be put to the places marked on the scheme.

The task was not easy. The containers were large and heavy; it was only possible to move them by pushing along the floor, and they were too heavy to push more than one of them at a time. In addition, the containers were so smooth that Stas could not pull or turn them; all he could do was to push them forward in front of him. The dimensions of the room corresponded to the size of containers exactly, so Stas could not clime over a container, squeeze himself between containers, or wriggle himself between a container and a wall. He could only move through unoccupied space. Thus, putting containers in order was a tricky puzzle. And if Stas could not solve it or put incidentally one of the containers into some corner from which it could not be extracted, then Stas was in real trouble. The point was that the walls of the room were solid, with no exits. In the morning, Stas got to the room through one of the hatches in the ceiling. He could not leave the room until the task was completed. When all containers were on their places, the smart control system opened a hatch with a rope-ladder for Stas right over him.

Help Stas to make a plan of moving the containers.

Input

You are given a scheme of the storeroom. This is a table of size $n \times m$ ($3 \le n, m \le 8$). An empty cell is shown by a space, and objects are denoted as follows:

- "#" is a piece of wall
- "." is an empty cell where a container must be put (an aim cell)
- "@" is the cell from which Stas starts his work if it is not an aim cell
- "+" is the cell from which Stas starts his work if it is an aim cell
- "$" is a container on a cell which is not an aim cell
- "*" is a container on an aim cell

It is guaranteed that the scheme of the room is correct, that is, Stas cannot go out of the room. The number of containers is equal to the number of aim cells.

Output

Output a plan of work for Stas. In a single line, you should specify his movements by letters r, l, u, and d, which correspond to the four possible directions of moves. If during a move a container is pushed, then the letters should be capital (R, L, U, and D, respectively). The string should be no longer than 10000 symbols. You may assume that there is a solution.

Examples

input.txt	output.txt
######## #@ $.# ########	rrRR
###### ## .# #@ ### # * # # $ # # # #######	dddrrrruLdlUUUluRR

Problem I. Bacon's Cypher

Input file:	input.txt
Output file:	output.txt
Time limit:	2 seconds
Memory limit:	64 megabytes

Programmer Vasya was down on his luck. Instead of a vacation, he was sent to a scientific conference.

"It is necessary to increase your competence," his boss said, "it's an important conference on cryptography, and it's held in France, where they used encryption in the days of de Richelieu and cracked codes in the days of Viete."

One of the talks at the conference was about the attempts to solve Bacon's cyphers. The speaker proposed a hypothesis that the key to Bacon's secrets could be found if all possible substrings of Bacon's works were analyzed.

"But there are too many of them!" Vasya expressed his astonishment.

"Not as many as you think," the speaker answered, "count them all and you'll see it yourself."

That evening Vasya found on the Web the complete set of Bacon's works. He wrote a program that converted the texts into one long string by removing all linebreaks, spaces, and punctuation marks. And now Vasya is confused because he doesn't know how to calculate the number of different substrings of this string.

Input

You are given a nonempty string consisting of lowercase English letters. The string is no longer than 5000 symbols.

Output

Output the number of different substrings of this string.

Example

input.txt	output.txt
aaba	8

Problem J. Abstract Thinking

Input file: input.txt
Output file: output.txt
Time limit: 2 seconds
Memory limit: 64 megabytes

This summer programmer Dima wanted to take a rest from programming because he felt that the level of abstraction his profession required was driving him mad. He decided to go to Greece. But he forgot that Greece was the homeland of geometry where famous Euclid lived and worked. In geometry, instead of real figures, abstract notions are studied, and the proofs are based not on intuition, but on axioms and formal definitions. Even shepherds in Greece have well-developed abstract thinking skills.

For example, consider the following problem, which will show the level of your abstract thinking abilities. Imagine a circle, then put mentally n points on its periphery at equal distances. After that connect (again mentally!) these points pairwise by straight segments (of course, you remember that such segments are called chords). Now consider any three different chords from this set that intersect pairwise. If at least one of their intersection points lies inside the circle, we will call the figure formed by these chords an interesting triangle. If you can count the number of interesting triangles correctly, then you can go to Greece and not be ashamed of yourself there.

Input

You are given the number n of points on the periphery of a circle ($3 \le n \le 2000$).

Output

Output the number of interesting triangles.

Examples

input.txt	output.txt
4	4
5	25

Problem K. Chinese Watches

Input file: input.txt
Output file: output.txt
Time limit: 2 seconds
Memory limit: 64 megabytes

When programmer Vova was in China he discovered that Russian watches "Zarya" were sold there ten times cheaper than in Russia. Vova decided to make some money and bought a lot of watches. He wanted to sell them at home at half-price (that would be five times more expensive than he had paid for them). But when he returned he found out that the watches showed different times; moreover, from a slightest push they stopped (or started to work again). Obviously, they were not real "Zarya" watches, but their exact copies. In order to sell the whole lot as quickly as possible, Vova wants to set them all at the same time. He then will be able to say that this is the time at the producing plant. Of course, before opening the suitcase he will have to push it a bit to make the watches start to work simultaneously.

In order to set a time on a watch, Vova must rotate the winder. He can make one turn of the winder in one second; as he does so, the second-hand makes a complete circle. When Vova turns the winder, the minute-hand rotates 60 times slower than the second-hand, and the hour-hand rotates 60 times slower than the minute-hand. For example, to set time six hours ahead it takes six minutes. The hands of the watches can be rotated clockwise only. Help Vova to prepare the watches for sale. Choose a time that should be set on all watches so that the total time Vova spends setting it is minimal.

Input

The first line contains the number n of watches Vova bought ($1 \le n \le 50000$). Then there are n lines. The $(i+1)$th line of the input contains the time that the ith watch shows, in the format `h:mm:ss`. Here the integer h ($1 \le h \le 12$) is the hour and the two-digit integers mm and ss ($00 \le mm, ss \le 59$) are the minutes and seconds, respectively.

Output

Output the time that should be set on all watches in the format given above.

Example

input.txt	output.txt
3 11:30:00 12:10:01 6:10:18	12:10:01