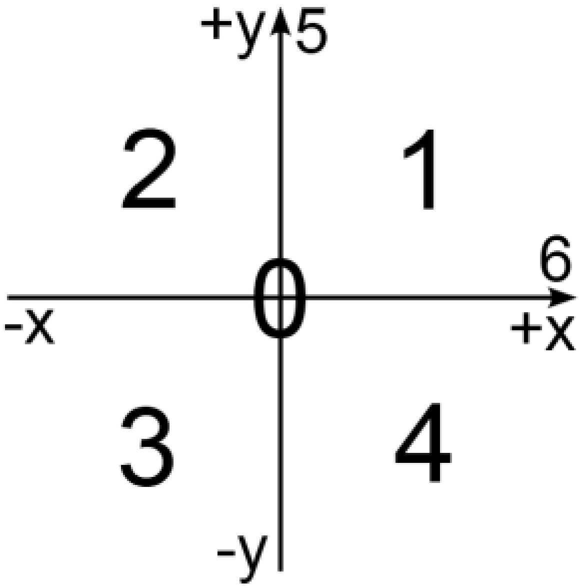
Telerik Software Academy – C# Fundamentals Part 1 – Sample Exam

Problem 1 – Cartesian Coordinate System

You are given a two-dimensional Cartesian coordinate system and the two coordinates (**X** and **Y**) of a point in the coordinate system. If you don't know what Cartesian coordinate system is Google it with Bing. As you will find, the coordinate system is divided by 2 lines (see the picture bellow) which divide the plain in four parts. Each of these parts has a lot of points that are numbered between 1 and 4. There is one point where our lines are crossing. This point has the following coordinates: X=0 and Y=0. As a result this point is numbered 0. The points on the lines are also numbered with the numbers 5 and 6 (again see the picture below).

Your task is to write a program that finds the number of the location of the given point in the coordinate system.



Input

Input data is being read from the console.

The number **X** is on the first input line.

The number **Y** is on the second input line.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output data must be printed on the console.

On the only output line you must print an integer number between 0 and 6, depending on the location of the given point in the coordinate system.

Constraints

- The numbers **X** and **Y** are numbers between -2 000 000 000 001 337 and 2 000 000 000 001 337, inclusive.

- Allowed working time for your program: 0.25 seconds.
- Allowed memory: 16 MB.

Examples

Input Example	Output Example
1 2	1
-3 -4	3
-3000 9000	2
12345 -98786543	4
1337 0	6

Problem 2 – Miss Cat 2011

There are two things that cats love most: 1) sleeping and 2) attending beauty contests. The most important thing for each female cat is the contest "Miss Cat". There are always ten cats that participate in the final round of the contest, numbered 1 to 10.

The jury of the contest consists of **N** people who subjectively decide which cat to vote for. In other words each person votes for just 1 cat that he has most liked, or from whose owner he has received the biggest bribe. The winner of the contest is the cat that has gathered most votes. If two cats have equal votes, the winner of the contest is the one whose number is smaller.

Your task is to write a computer program that finds the number of the cat that is going to win the contest "Miss cat"

Input

The input data is being read from the console.

The number **N** is on the first input line.

An integer between 1 and 10 is written on each of the next **N** lines (this is the number of the cat)

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output data must be printed on the console.

On the only output line you must print the number of the cat, which has won the contest.

Constraints

- The number **N** is a positive integer between 1 and 100 000, inclusive.
- The numbers of the cats for which the jury votes are positive integer numbers between 1 and 10, inclusive.
- Allowed working time for your program: 0.25 second.
- Allowed memory: 16 MB.

Examples

Input example	**Output example**
1 6	6
4 1 3 3 7	3
5 1 2 3 1 2	1

Problem 3 – Forest Road

Geeko, a non-stop learning trainee at Telerik Software Academy lived deep into the Lulin forests. Every time he went to the Academy he had to take a long trip through the forest. Starting from the top left corner of the forest, the road always goes down and right first and when it reaches the border, it goes down and left.

The Academy is situated in the bottom left corner, and Geeko begins his journey from the top left corner of the forest (see the examples below).

He wanted to make a program that generates a map of the forest but he couldn't. Help Geeko on his way to the Academy by writing the program instead of him.

Input

The input data is being read from the conso le.

On the only line in the console you are given an integer number **N**, showing the width of the map. The map's height is always **2*N - 1**.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output data must be printed on the console.

You should print the whole map on the console. Use the symbol "*****" (asterisk) to mark Geeko's path and "**.**" (dot) to illustrate the trees.

Constraints

- The number **N** is a positive integer between 2 and 79, inclusive.
- Allowed working time for your program: 0.25 second.
- Allowed memory: 16 MB.

Samples

Sample input	Sample output

4	`*...`
	`.*..`
	`..*.`
	`...*`
	`..*.`
	`.*..`
	`*...`

5	`*....`
	`.*...`
	`..*..`
	`...*.`
	`....*`
	`...*.`
	`..*..`
	`.*...`
	`*....`

Problem 4 – Binary Digits Count

You are given a sequence of **N** positive integer numbers and one binary digit **B (0 or 1).** Your task is to write a program that finds the number of binary digits (**B**) in each of the **N** numbers in binary numeral system. Example: 20 in the binary numeral system looks like this: 1**0**1**00**. The number of binary digits 0 of the number 20 in the binary numeral system is 3.

Input

The input data is being read from the console.

On the first input line there will be the digit **B.**

On the second line you must read the number **N**.

On each of the following **N** lines there is one positive integer number written – the consequent number, whose sum of binary digits **B** we are searching for.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output must be printed on the console.

In the output you must have **N** lines. Each line must have 1 integer number – the number of digits **B** in the binary representation of the given consequent number.

Constraints

- Number **N** is a positive integer between 1 and 1000, inclusive.
- Each of the **N** numbers is a positive integer between 1 and 4 000 000 000, inclusive.
- The digit **B** will be only 0 or 1.
- Allowed work time for your program: 0.25 second.
- Allowed memory: 16 MB.

Examples

Input Example	**Output Example**
1 10 1 2 3 4 5 6 7	1 1 2 1 2 2 3 1 2

8 9 10	2
0 4 20 1337 2147483648 4000000000	3 5 31 19
0 6 1 4 16 64 256 1024	0 2 4 6 8 10

Problem 5 – Subset Sums

You are given a list of **N** numbers. Write a program that counts all non-empty subsets from this list, which have sum of their elements exactly **S**.

Example: if you have a list with 4 elements: { 1, 2, 3, 4 } and you are searching the number of non-empty subsets which sum is 4, the answer will be 2. The subsets are: { 1, 3 } and { 4 }.

Input

The input data is being read from the console.

On the first input line there will be the number **S.**

On the second line you must read the number **N**.

On each of the following **N** lines there will be one integer number written – all the numbers from the list.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output must be printed on the console.

On the only output line you must print the number of the non-empty subsets, which have sum of all its elements exactly **S**.

Constraints

- The number **N** is a positive integer between 1 and 16, inclusive.
- All of the **N** numbers are integer numbers and will be between -1 337 000 000 000 and 1 337 000 000 000, inclusive.
- The number **S** is an integer number between -21 392 000 000 000 and 21 392 000 000 000, inclusive.
- All of the **N** numbers will be distinct.
- Allowed work time for your program: 1 second.
- Allowed memory: 16 MB.

Examples

Input Example	**Output Example**
1 1 1	1
0 5 -2 -1 1 2 3	4
1337 4 12 23 34 45	0