Problems Overview

Problem A: Chat log analysis

Problem B: Fake Fibonacci numbers

Problem C: Robots

Problem D: M-drugs

Problem E: Expansion

Problem F: Signature

Problem G: Goldbach's conjecture

Problem H: Nice numbers

Problem I: Rendezvous

Problem J: City Lights

Problem K: Shuffling cards

Note: The input and output for all the problems are standard input and output.

Problem A: Chat log analysis

A scientist wants to analyse the emotions of chatters by extracting text-based emoticons within a chat log, which is a text string. He considers four emoticons and assigns them different values for each level of happiness:

:(	a colon followed by an open parenthesis means sad. It is assigned the value -1.
:((	a colon followed by two open parentheses means very sad. It is assigned the value -5.
:)	a colon followed by a close parenthesis means happy. It is assigned the value 2.
:))	a colon followed by two close parentheses means very happy. It is assigned the value 4.

In order to calculate the level of happiness for a chat, the scientist would like to extract all emotional symbols and the happiness level of a chat is the sum of the value of the emotional symbols in the chat. Your task is to write a program to help him to do so.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

Each data set consists of only one line, which contains the text string representing the chat log.

Output

For each data set, write in one line the level of happiness of the chat log.

Sample Input	Sample Output
3 I am happy :), very happy :)). I am happy :(, no actually sad :((I win :), but I lost some friends :(:(, kidding :))	6 -6 4

Problem B: Fake Fibonacci numbers

Mr. X invented a new way to create a sequence of numbers which he called "Fake Fibonacci numbers" because the rule to create the sequence is very similar to the rule for Fibonacci numbers. Mr. X's sequence starts with some four initial numbers S_1, S_2, S_3, and S_4 ($1 \le S_1, S_2, S_3, S_4 \le 5$).

A number S_i ($i > 4$) in the sequence is the sum of S_{i-4} and S_{i-1}:

$$S_i = S_{i-4} + S_{i-1}$$

For a given odd positive integer **N**, Mr. X would like to generate the first **N** fake Fibonacci numbers, then sort them in ascending order, and find out what the middle number of the sorted sequence is. For example, with S_1=3, S_2=2, S_3=4, S_4=1, and **N = 7**, the first 7 fake Fibonacci numbers are:

3, 2, 4, 1, 4, 6, 10

After sorting, the sequence is:

1, 2, 3, 4, 4, 6, 10

And the middle number of the sorted sequence is 4.

Given four initial numbers S_1, S_2, S_3, and S_4, and **N**, your task is to write a program to help Mr. X find the middle number in the sorted sequence of the first **N** fake Fibonacci numbers.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

Each data set consists of only one line, which contains an odd integer **N (5 ≤ N ≤ 19)** followed by four numbers S_1, S_2, S_3, and S_4 separated by space.

Output

For each data set, write in one line the middle number of the sorted sequence of the first **N** fake Fibonacci numbers.

Sample Input	Sample Output
2 5 3 2 4 1 7 3 2 4 1	3 4

Problem C: Robots

In preparation for the ACM/ICPC in Hanoi in 2020, a huge amount of robots is needed to serve as volunteers. However, the organizers own only a certain number of robots. Fortunately, these robots can automatically duplicate themselves. More precisely, a group of **R** robots together in **T** time units can build a new robot like themselves. The new robot also has the ability to join a group of robots to build another robot. Each robot can join only one group for building new robots at a time.

For example with **R**=4 and **T**=6, a group of 4 initial robots takes 6 time units to build the 5^{th} robot, another 6 time units to build the 6^{th} robot, another 6 time unit to build the 7^{th} robot, and another 6 time unit to build the 8^{th} robot. After that, two groups can be formed to build new robot concurrently, which take only 6 time unit to build 2 new robots. So it takes 30 time units to have 10 robots from 4 initial robots.

Given 3 positive integers **R, T** and **N** ($1 \le R \le 1024$, $1 \le T \le 10^4$, $R < N \le 10^9$), your task is to write a program to determine the smallest time units required to build a team of **N** robots from **R** initial robots.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 30. The following lines describe the data sets.

Each data set consists of only one line that contains three integers **R, T** and **N** separated by a space.

Output

For each data set, write in a single line the smallest time unit required.

Sample Input	Sample Output
2 4 6 10 4 6 11	30 36

Problem D: M-drugs

Amino acids are the basic elements of living cells. There are 20 different types of amino acids which are denoted by capital letters: A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, and Y. A protein is a chain of amino acids.

Viruses have been well known for their causes for many diseases. A virus is characterised by its protein. Recently, they have invented M-drug (molecular drug) - a new type of medicine to kill virus, which has shown its surprising efficiency. An M-drug is also characterised by its protein.

An M-drug characterised by a protein *D*, can destroy *k* amino acids in the protein *V* of a virus if there exists two number sequences ($x_1 < x_2 <\ldots< x_k$) and ($y_1 < y_2 <\ldots< y_k$) satisfying that $D(x_i) = V(y_i)$ with $i = 1\ldots k$. The efficiency of an M-drug on a virus is defined as the maximum number of amino acids in the protein of the virus that the M-drug can destroy.

Given the protein of a virus and the proteins of ***n*** different M-drugs, your task is to write a program to determine the maximum value of the efficiencies of those M-drugs on the virus.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

For each data set, the first line contains an integer ***n*** ($1 \le n \le 200$) representing the number of M-drugs. The next line consists of a string with its length not exceeding 10000 representing protein of the virus. The i^{th} line of the following ***n*** lines contains a string with a length not exceeding 500 representing the protein of the i^{th} M-drug.

Output

For each data set, write in one line the maximum value of the efficiencies of those M-drugs on the virus.

Sample Input	Sample Output
1 5 ENRPPNVPES TEV LNRC HKVR FWW PWP	2

Problem E: Expansion

There are N countries exploring a new discovered planet, which is represented as a grid in a Cartesian plane. At the beginning, each country occupies a separate (non-overlapping) territory. The territory has the shape of a rectangle whose edges are parallel to the axes Ox and Oy. The territory is determined by four numbers x_1, y_1, x_2, y_2 where (x_1, y_1) is the coordinate of the bottom left corner and (x_2, y_2) is the coordinate of the top right corner of the rectangle ($10^{-8} \le x_1, y_1, x_2, y_2 \le 10^8$). Every month, each country expands its territory to one of four directions: **left** - the new territory is determined by x_1-1, y_1, x_2, y_2; **right** - the new territory is determined by x_1, y_1, x_2+1, y_2; **up** - the new territory is determined by x_1, y_1, x_2, y_2+1; and **down** - the new territory is determined by x_1, y_1-1, x_2, y_2. The conflict happens between two countries when their territories start to overlap each other (the area of the overlapped region is non-zero). Given the plan of expansion of these countries for the next T months, your task is to write a program to determine after how many months, the first conflict will happen.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

For each data set, the first line contains two integers N and T ($N \le 100$, $T \le 100\,000$). For the i^{th} pair of consecutive lines of the following N pairs of consecutive lines, the first line contains four integers x_1, y_1, x_2, y_2 separated by a space determining the initial territory of the i^{th} country; and the second line consists of a string of length T only containing 'L', 'R', 'U', 'D' characters describing the plan of expansion of the i^{th} country for the next T months where 'L' means left, 'R' means right, 'U' means up, and 'D' means down.

Output

For each data set, write in one line the number of months after which the first conflict will happen, or -1 if no conflict happens.

Sample Input	Sample Output
2 3 6 1 1 2 2 LLURRR 1 3 2 4 LLLLLL 4 1 5 2 LUUUUU 3 10 1 1 2 2 LLLLLLLLLL 1 3 2 4 RRRRRRRRRR 4 1 5 2 DDDDDDDDDD	5 -1

Problem F: Signature

A digit of a decimal integer ***N*** is a local maximum if it is bigger than both of its adjacent digits. For simplicity, the first and last digits of a number are not local maxima.

Let us consider a signature of a decimal number ***N*** which is defined as the number of local maximum digits of that number. Your task is to write a program to compute the sum of signatures of all numbers in a range from **X** to **Y** inclusively.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 50000. The following lines describe the data sets.

Each data set consists of only one line, which contains two integers **X** and **Y** separated by a space **($0 \le X \le Y \le 10^{15}$)**.

Output

For each data set, write in one line the sum for the signatures of all numbers in the range from **X** to **Y** inclusively.

Sample Input	Sample Output
4 1 100 121 121 909 909 100 150	0 1 0 10

Problem G: Goldbach's conjecture

You must have heard of Goldbach's conjecture, a well-known unsolved problem in number theory. It is stated that every even integer greater than 2 can be written as a sum of two prime numbers. Simple, yet extremely hard! No mathematician has been able to prove this conjecture for nearly 300 years. For example:

```
 4 = 2 + 2
 6 = 3 + 3
 8 = 3 + 5
10 = 3 + 7
   = 5 + 5
12 = 5 + 7
```

```
14 = 3 + 11
   = 7 + 7
16 = 3 + 13
   = 5 + 11
18 = 5 + 13
   = 7 + 11
...
```

Let G(N) be the number of different ways to represent a number of the form 2N as a sum of two prime numbers. As we have seen in the above examples:

`G(i) = 1, 1, 1, 2, 1, 2, 2, 2` for `i = 2..9`

With the definition of G(***N***), the Goldbach's conjecture can be stated as follows: G(N) > 0 for all positive integers N > 1.

Given a number N, your task is to write a program to compute the following sum:

`F(N) = G(2) + G(3) + ... + G(N)`

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

Each data set consists of only one line which contains an integer N ($3 \le N \le 500000$).

Output

For each data set, write in a single line the sum F(N).

Sample Input	Sample Output
3 7 4 9	8 3 12

Problem H: Nice numbers

A "nice number" is a number that starts with some repeating sequences 123456789 and followed by some additional trailing zeroes:

123456789…1234567890..0

Given a positive integer *N*, the question is whether there is a multiple of *N* which is a nice number? For example, for *N* = 342 we have:

342*3609847635188795 = 1234567891234567890

Given an integer *N*, your task is to write a program that will find the smallest multiple of *N* which is a nice number or determine that no such one exists.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

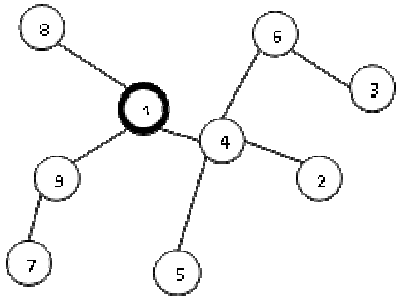
Each data set consists of only one line which contains an integer *N* ($1 < N < 10^6$).

Output

For each data set, write in one line the smallest multiple of *N* which is a nice number, or -1 if no such number exists.

Sample Input	Sample Output
1 342	1234567891234567890

Problem I: Rendezvous



Treeland Security Agency (TSA) is a secret organization supposed to maintain the security in **N** cities of Treeland, which are identified by the numbers from 1 to **N.** The city with identifier 1 is the headquarters of TSA. For the safety of all agents working for TSA, to move from one city to another city one must follow a designated two-way road system which has a tree structure rooted at the headquarters. For every mission of TSA, two secret agents located in two different cities are assigned. After completing the mission, they have to gather at a designated city, called rendezvous, and travel together to the headquarters. The rendezvous is a city which appears in both agents' shortest routes to the headquarters and is farthest from the headquarters. This year, the director of TSA has planned **K** missions that will be assigned to **K** pairs of secret agents. He is quite interested in finding the most popular rendezvous of the year, which is the rendezvous for the largest number of missions. Given a road system among **N** cities and a list of cities assigned to agents in **K** missions, your task is to write a program to determine the most popular rendezvous.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 25. The following lines describe the data sets.

For each data set, the first line contains two integers **N** (**N** ≤ 100 000) and **K** (**K** ≤ 100 000). The i^{th} line of the following **N**-1 lines contains two integers **u** and **v** (**1** ≤ **u**,**v** ≤ **N**) separated by a space indicating that there is a two-way road between city **u** and **v**. The j^{th} line the next **K** lines contains two integers **x** and **y** (**1** ≤ **x**,**y** ≤ **N**) separated by space describing the two cities assigned to two secret agents in the j^{th} mission.

Output

For each data set, write in one line the identifier of the city which is the most popular rendezvous. In case there are more than one solution, write the smallest identifier.

Sample Input	Sample Output
1 9 3 1 8 1 4 1 9 9 7 4 6 6 3 4 2 4 5 5 3 2 6 7 8	4

Problem J: City Lights

Down town BigCity is covered by a grid of ***n*** East-West avenues (numbered from 1 to ***n***) intersecting with ***n*** North-South streets (numbered from 1 to ***n***). At the intersection between an avenue and a street, there is a light post. At each avenue or street, there is a power switch which turns on all ***n*** lights along that avenue/street. At day time, all the lights are automatically turned off. Every evening, due to security reasons, the lights at a certain number of intersections must be turned on. To minimize the effort to turn on the required lights, they wish to use the smallest number of power switches.

Your goal is to write a program that computes the smallest number of power switches that must be used to turn on the required lights.

Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 200. The following lines describe the data sets.

For each data set, the first line contains an integer ***n*** ($1 \le n \le 30$). Next ***n*** lines (each of which contains ***n*** binary digits separated by space) contains a matrix ***R*** of size ***n***×***n*** representing the requirements about the lights which need to be turned on. The value 1 of ***R***(i,j) means that the light at the intersection between i^{th} avenue and j^{th} street must be turned on. The value 0 of ***R***(i,j) means that there is no requirement for the light at the intersection between i^{th} avenue and j^{th} street.

Output

For each data set, write in one line an integer representing the smallest number of power switches that must be turned on.

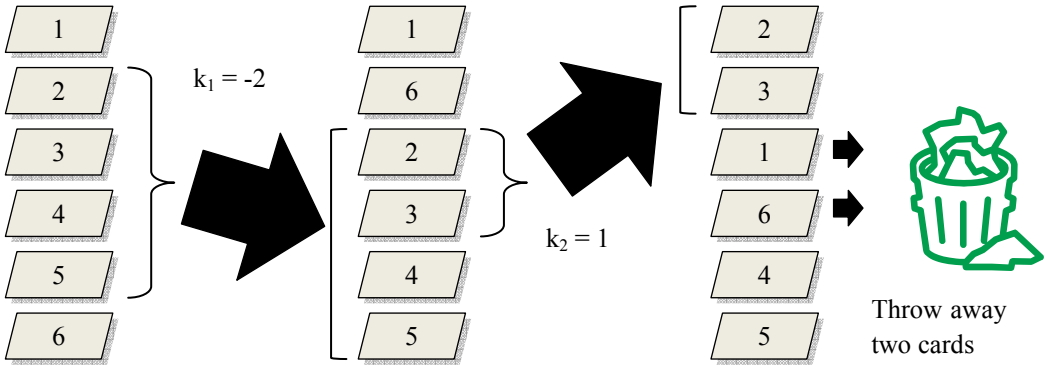
Sample Input	Sample Output
1 7 0 0 1 1 0 1 1 1 0 1 0 0 0 1 1 0 1 0 0 0 1 0 1 0 0 1 1 0 1 0 1 0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1	5

Problem K: Shuffling cards

A deck of 2N cards with distinct values of 1,2, …, 2N is given to the shuffling machine. At the beginning, the cards are arranged in the deck in ascending order from the top to the bottom. The shuffling machine executes a sequence of M instructions determined by M integers $k_1, k_2, \ldots, k_M$ for shuffling the cards. The instruction determined by the number k_i ($1 \le |k_i| < N$), commands the machine to shuffle the cards as follows:

- If $k_i > 0$: remove a pile of $2k_i$ cards at the middle of the deck and stacks them on top of the deck.
- If $k_i < 0$: remove a pile of $-2k_i$ cards at the middle of the deck and inserts them into bottom of the deck.

Mr. X received the deck after it has been shuffled according to M instructions. He wants to throw away some cards from the deck in such a way that the values of the remained cards are in an increasing order from the top to the bottom. Given the M instructions for the shuffling machine, your task is to write a program to help Mr. X determine the minimum number of cards to be removed after the deck has been shuffled.



Input

The input file consists of several data sets. The first line of the input file contains the number of data sets which is a positive integer and is not bigger than 20. The following lines describe the data sets.

Each data set consists of two lines. The first line contains two positive integers N and M ($2 \le N \le 10^9$; $0 \le M \le 10^5$) separated by a space. The second line contains M integer $k_1, k_2, \ldots, k_M$ separated by a space.

Output

For each data set, write in one line the minimum number of cards to be removed after the deck has been shuffled.

Sample Input	Sample Output
2 3 2 -2 1 1000000000 3 999999999 -1 2	2 7