AIPO 2026
Preliminary Round

January 10, 2026

AIPO

All Ireland Programming Olympiad

UCC

University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Contributors

The AIPO organisers would like to thank the following people.

For leading the question writing:

- Dr Sabin Tabrica
- Andrew Nash

Time limits and input/output bounds will become available when the submission system opens.

1 Add Two Numbers

This problem is designed to make sure you understand how to use our submission system. If you have trouble with this task, first check out our Tips & Tricks document available at aipo.ucc.ie. Failing that, please contact anash@umail.ucc.ie

You are given 2 integers a, b as standard inputs to your program. Print the sum a+b on a single line.
A tutorial for this problem will be available on our website https://aipo.ucc.ie. This problem is intended to be a simple test to make sure you are able to understand the operations of the submission server.

Input/Output

Input

Two space-separated integers a, b on a single line.

Output

A single integer on a single line

Constraints

$-1,000,000 \leq a, b \leq 1,000,000$

Examples

Sample Input 1

```
3 2
```

Sample Output 1

```
5
```

Sample Input 2

```
-5 5
```

Sample Output 2

```
0
```

Sample Input 3

```
1000 -54391
```

Sample Output 3

```
53391
```

2 Mirror Pairs

The mirror of a natural number x is the number obtained by going through the digits of x from right to left, ignoring the zero digits in the last positions of x. For example, the mirror of 123 is 321, while the mirror of 1200 is 21. A natural number x is considered a palindrome if x is equal to its mirror. For example, the number 12321 is a palindrome.

Two numbers can be formed from two distinct numbers by joining one to the right of the other. For example, from the numbers 123 and 32, the numbers 12332 (from adding 32 to the right of 123) and 32123 (from adding 123 to the right of 32) can be obtained.

Requirements

Let there be a sequence of natural numbers $a = (a[i], i = 0, \ldots, n - 1)$.

Determine the largest palindromic number that can be formed by joining two distinct numbers in the sequence.

2.1 Input Data

The input will have 2 lines:

- The first line is the natural number n.
- The second line contains the list of natural numbers $a = (a[i], i = 0, \ldots, n - 1)$ separated by a space.

2.2 Output Data

The output will contain a single number, representing the corresponding result for the given requirement.

2.3 Constraints

- $1 \leq n \leq 100000$.
- $1 \leq a[i] < 10000$

Examples

Sample Input

```
6
24 79 42 97 123 124
```

Sample Output

```
42124
```

Explanation The following palindrome numbers can be formed: 2442, 4224, 7997, 9779, and 42124. The largest of these is 42124.

3 Primes with Distinct Digits

When John played with prime numbers, he observed that most are made with distinct digits. For example, 13 or 17 are made with distinct digits however, 11, 131 or 11311 are primes in which some digits occur multiple times. Then John started to count how many such prime numbers are less than 100 and he managed to find 24 primes, however he could not find how many are less than 1000.

3.1 Requirements

You need to write a program to find how many primes with distinct digits are less than a given number n.

3.2 Input Data

The input will have one line to contain the number n.

3.3 Output Data

The output will contain the number of such primes.

3.4 Constraints

n has maximum 10 digits.

Examples

Sample Input 1

```
10
```

Sample Output 1

```
4
```

Sample Input 2

```
100
```

Sample Output 2

```
24
```

Sample Input 3

```
200
```

Sample Output 3

```
38
```

Explanation The primes with distinct digits for each example are:

Case	Primes with distinct digits
Sample 1	2, 3, 5, 7
Sample 2	2, 3, 5, 7, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97
Sample 3	2, 3, 5, 7, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 103, 107, 109, 127, 137, 139, 149, 157, 163, 167, 173, 179, 193, 197

4 Problem: Firewall Breach

Description

An elite hacker is attempting to extract data from a facility consisting of two vertical parallel mainframes, each of height n segments. The segments are numbered from 1 to n from bottom to top.

Some segments are **Open** (represented by '-') and others are **Locked** (represented by 'X'). The hacker cannot occupy a Locked segment.

Initially, at time $t = 0$, the hacker is located at segment 1 of the **left mainframe**. Every second, the hacker must perform exactly one of the following actions:

- **Climb up:** Move to segment $x + 1$ on the current mainframe.
- **Climb down:** Move to segment $x - 1$ on the current mainframe.
- **Cross-link:** Jump to segment $x + k$ on the **opposite** mainframe.

Simultaneously, a security "purge" protocol is active. The purge starts at the bottom and rises by 1 meter every second. After t seconds (where t is the number of actions performed), all segments $x \leq t$ are wiped. The hacker cannot be on a segment that has been purged.

The movement and purge happen in cycles: first, the hacker performs an action, then the purge level increases.

The hacker successfully escapes if they reach a segment with a height strictly greater than n. Determine if it is possible to complete the escape.

Input

The first line contains two integers n and k ($1 \leq n, k \leq 10^5$)

Examples

Sample Input 1

```
7 3
---X--X
-X--XX-
```

Sample Output 1

```
YES
```

Sample Input 2

```
6 2
--X-X-
X--XX-
```

Sample Output 2

```
NO
```

5 Wedding Shopping (Taken from Competitive Programming 4)

Foreword

The following task is an existing problem that is worked through in the textbook Competitive Programming 4 by Stephen and Felix Halim, which is an excellent tool for learning competitive programming. If you do not understand how to solve this problem, I strongly recommend that you read the take the time to read, understand, and learn from this textbook, as well as other resources on Dynamic Programming. If you simply download an existing solution, or use the assistance of an LLM for these tasks you are wasting nobody's time and opportunity but your own, as this will not help you get any closer to the AIPO final.

Problem Statement, Credit UVa Online Judge

One of our best friends is getting married and we all are nervous because he is the first of us who is doing something similar. In fact, we have never assisted to a wedding, so we have no clothes or accessories, and to solve the problem we are going to a famous department store of our city to buy all we need: a shirt, a belt, some shoes, a tie, etcetera. We are offered different models for each class of garment (for example, three shirts, two belts, four shoes, ...). We have to buy one model of each class of garment, and just one. As our budget is limited, we cannot spend more money than it, but we want to spend the maximum possible. It's possible that we cannot buy one model of each class of garment due to the short amount of money we have.

Input

The first line of the input contains an integer, N, indicating the number of test cases. For each test case, some lines appear, the first one contains two integers, M and C, separated by blanks ($1 \leq M \leq 200$, and $1 \leq C \leq 20$), where M is the available amount of money and C is the number of garments you have to buy. Following this line, there are C lines, each one with some integers separated by blanks; in each of these lines the first integer, K, ($1 \leq K \leq 20$), indicates the number of different models for each garment and it is followed by K integers indicating the price of each model of that garment

Output

For each test case, the output should consist of one integer indicating the maximum amount of money necessary to buy one element of each garment without exceeding the initial amount of money. If there is no solution, you must print 'no solution'.

Examples

Sample Input

```
3
100 4
3 8 6 4
2 5 10
4 1 3 3 7
4 50 14 23 8
20 3
3 4 6 8
2 5 10
4 1 3 5 5
5 3
3 6 4 8
2 10 6
4 7 3 1 7
```

Sample Output

```
75
19
no solution
```