CODEx 2019

Instructions:

- The following set contains 10 Questions
- Each question consists of the following
    - Problem Statement
    - Input Format
    - Output Format
    - Constraints
    - Test Case for Input
    - Test Case for Output
    - Explanation (only where required)

- Each question will be judged on 10 test cases and points will be allocated on the number of test cases passed
- You need to pass a minimum of 60% i.e. 6 test cases in order to gain partial points
- Please save all your code files in a folder named as your team code.
- Please write your code accurately, accuracy will matter the most.
- **The questions are in NO particular order of difficulty**
- Please check that there are 8 questions with 17 pages in total
- You will be allotted 2 Hours for the competition.

Problem A

PASS OR FAIL

Keanu Reeves is very sad because he has his maths exam tomorrow and he hasn't studied for it. So he comes up with a perfect strategy, to answer *'YES'* to every question asked. Being a problem setter for the exam and his dear friend (we mean YOU) decides to help him out while he goes to eat his hamburger on a bench.

The question is of the following format. You are given an integer *N*, you have to tell whether the following integer can be represented as a power of another integer *x*, i.e. *N* can be represented as x^y where *y* is some integer >1.

Given that the total number of questions are *T*, Tell in how many of these questions Keanu will have to change his answer to *'NO'*, so as to score full marks.

INPUT

- The first line of input contains an integer *T* which denotes the number of questions in the exam.
- This is followed by *T* lines, each of which contain a single integer *N*, as mentioned above.

OUTPUT

- Print a single integer, the number of answer Keanu needs to change.

CONSTRAINTS

- 1 <= *T* <= 100
- 2 <= *x* <= 10^5
- 2 <= y <= 10^5

SAMPLE INPUT

3
8
49
10

SAMPLE OUTPUT

1

EXPLANATION

$2^3 = 8$
$7^2 = 343$
But 10 cannot be represented as x^y for any integer positive x and y.

Problem B

STORMING AREA 51

It's 20th September and following the hype you have reached Area 51 with you squad. Upon entering through the first level of Area 51 you are greeted with a 100ft tall wall and to reach the second level, you will need to cross the wall. There is a door which leads you to the second level but it is encoded with a passcode. Unlike traditional passcodes, this passcode tests your intelligence and requires you to answer the given question.

However, ignoring the secret gate, your squad tries to find other methods to pass the wall. The sun is about to set and you don't have a lot of time left in your hand. Being the leader of your squad, you decide to take the responsibility of giving the answer to the question yourself. The question is as follows. Giving a list *A* of *N* integers, let *X* be the **XOR** of the integers. You have to tell the **minimum** number of integers you have to add in the array so as to make *X* i.e. the xor of the array equivalent to **0** and input these numbers in the panel of the gate. If your answer is right then your team can pass the gate otherwise the gate will set an alarm which would alert the security forces leading to your execution.

INPUT

- The first line of the input contains a single integer *T* denoting the number of test cases
- The first line of each test case contains a single integer *N* denoting the number of integers in the list *A*
- The second line of each test case contains *N* spaced integers: A_1 A_2 A_3 ….. A_N denoting the elements of the list *A.*

OUTPUT

- For each test case, output a single integer *M*, in a new line, denoting the number of integers that need to be added to *A* to make xor of all its elements equal to 0.

CONSTRAINTS

- $1 <= T <= 100$
- $1 <= N <= 10^5$
- $1 <= A_i <= 10^5$
- The sum of *N* over all test cases does not exceed 10^5

SAMPLE INPUT

```
1
4
5 5 5 5
```

SAMPLE OUTPUT

```
0
```

Problem C

300IQ PLAY

You and your friend decide to play a game using a stack consisting of *N* bricks. In this game, you can alternatively remove 1, 2 or 3 bricks from the top, and the numbers etched on the removed bricks are added to your score. You have to play so that you obtain the maximum possible score. It is given that your friend will also play optimally and you make the first move.

As an example, bricks are numbered **arr = [1,2,3,4,5]**. You can remove either [1] = 1, [1,2] = 3 or [1,2,3] = 6. For your friend, your moves would leave the options of 1 to 3 elements from [2,3,4] = 9 leaving 5 for you (total score = 6), [3,4,5] = 12 or [4,5] = 9. In this case, it will never be optimal for your friend to take fewer than the maximum available number of elements. Your maximum possible score is **6**, achievable two ways: 1 first move and 5 the second, or [1,2,3] in your first move.

INPUT

- The first line of each test case contains a single integer *T* denoting the number of test cases.
- The first line of each test case contains a single integer *N* denoting the number of bricks
- The next line contains *N* spaced integer A_1 A_2 A_3 …… A_N denoting the score (i.e. value etched on removing) of that brick.

OUTPUT

- For each test case, print a single line containing your maximum score.

CONSTRAINTS

- 1 <= T <= 5
- 1 <= N <= 10^5
- 0 <= A_i <= 10^9

SAMPLE INPUT

```
2
5
999 1 1 1 0
5
0 1 1 1 999
```

SAMPLE OUTPUT

```
1001
999
```

EXPLANATION

In the first test case, you will pick 999,1,1. If you play in any other way, you will not get a score of 1001. In the second case, the best option will be to pick up the first brick (with 0 score) at first. Then your friend will choose the next three blocks, and you will get the last brick.

Problem D

<u>GRUMPY CAT</u>

Grumpy Cat has been given some maps to color.

Grumpy Cat can stand on two legs; meaning it can use the other two paws to color the maps. But the cat is too lazy to clean his paws after coloring a country. So, he can use at most two colors to color each map.

The maps must be colored keeping in mind the fact the if two countries share their borders, then they must not have the same color. Grumpy cat knows that some maps can be colored using only two colors and some cannot. If he colors a map partly, only to realize that the given map cannot be colored using only two colors, then he will get even grumpier!

Grumpy Cat now seeks your help to determine beforehand whether the given map can be colored using only two colors. He will give you the list of all neighboring countries in the map (i.e. the countries that share their borders.). You have to output 'YES' or 'NO', depending on whether the given graph can be colored using only two colors.

INPUT :

- First line of input contains a number M denoting the number of maps Grumpy Cat has to color. Then description of each map follow.
- For every map M, you'll be given a number N, denoting the number of shared boundaries on the map and a number C denoting the number of countries. Then N lines follow.
- Each of those N lines will have two space separated integers, A and B, denoting that countries A and B have a shared boundary.

OUTPUT :

- For every map M, output 'YES' (without quotes) or 'NO' (without quotes) in a new line

CONSTRAINTS

- 1 <= M <= 10
- 1 <= N <= 5000
- 1 <= A,B,C <= 50

SAMPLE INPUT

```
1
2 3
1 2
2 3
```

SAMPLE OUTPUT

```
YES
```

Problem E

<u>NOT MATURE YET</u>

Congratulations its your 18th Birthday Today ! Yay ! And you start feeling like a grown up already. You think you've turned mature and are finally responsible enough to take your own decisions. However, your friends know you better than you do yourself and they know you are not mature enough to take your own decisions and stand on your own feet, thus, they have all collected money and decided to buy you a mature bag as your birthday gift. The only catch is that they are themselves not mature yet and have thus lost it on their way to your home.

Let's define a ***NxN*** maze which begins at your house i.e. *(0,0)* [top left] and ends at *(N,N)* [bottom, right) denoting the coordinates of the place where they forgot your birthday present. You decide that you will find your gift whatsoever, so you set out to conquer this maze. Let *(i,j)* denote your position at any given instant, you are only allowed to go to the cells *(i+1,j)* or *(i,j+1)* in the next move provided that you remain within the maze and do not go out of it. However, you are not allowed to visit all the cells as some cells are locked. These require a **key** to go into. Let **1** represent the value of those cells which require a key to go into. Given a **single** key (i.e. you can visit only one locked cell on your path), you have to tell whether it is possible to get your birthday gift back or not.

INPUT

- The first line of the input contains a single integer *N* denoting the side length of the maze.
- The next N lines containing *N* spaced integers A_1 A_2 A_3…… A_N each of which have either '0' or '1' as their value. '0' denotes that a key is not needed to go into the particular cell, '1' denotes that a key is needed to enter that cell.

OUTPUT

- Print a single line for each test case. Print *'YES*' (without quotes) if it possible to reach the end of the maze with a single key else print '*NO'* (without quotes), if it is not possible to reach the end with a single key.

CONSTRAINTS

- 1 <= N <= 10^5
- A_i is either 0 or 1

SAMPLE INPUT

```
2
0 0
1 0
```

SAMPLE OUTPUT

```
YES
```

PROBLEM F

LAZY HACKER

Motivated by Digital India campaign you decide to become a hacker- A Black Hat hacker. However already being evicted two times, you finally decide to reform your life and become a white hat hacker. However, seeing the incentives of being a black hat hacker you are not so sure about your decision. You will only work if you earn a minimum of ***B*** dollars every day. But you are lazy, you cannot work more than **12 hours** a day, even if your work is making memes. You are a freelancer and work earn money as a bug bounty digger, i.e. you work with different companies and help them find bugs in their systems. You are a pro in finding bugs and can tell how much time will it take to discover one even before finding it. Let's say a bug awards you $\boldsymbol{M_i}$ units of money and takes $\boldsymbol{S_i}$ time to find and document, you have to tell the maximum amount of money you can earn that day.

INPUT

- The first line contains a single integer *T* denoting the number of test cases.
- The first line of each test case contains a single integer *N* denoting the number of bugs.
- The second line of each test case contains *N* spaced integers M_1 M_2 M_3 … M_N where M_i denotes the amount of money which is earned by the i^{th} bug
- The third line of each test case contains *N* spaced integers S_1 S_2 S_3…. S_N where S_i denotes the time taken to discover and document the i^{th} bug.

OUTPUT

- Print a single line containing an integer, denoting the maximum amount of money that can be made in a maximum of 12 hours.

CONSTRAINTS

- $1 <= T <= 100$
- $1 <= N <= 10^5$
- $1 <= M_i <= 100$
- $1 <= S_i <= 12$

SAMPLE INPUT

```
1
3
4 7 4
7 11 2
```

SAMPLE OUTPUT

```
8
```

EXPLANATION

We will choose the A1 = 4 and A3 = 4 for a total of 8 units of money in 9 hours. This is the maximum amount of money that can be earned

PROBLEM G

Area 51 In and Beyond

Bravo!! You cleared the puzzle and got your way into the second level of Area 51. You are now being applauded as a hero by your squad members. However, even before you could realise with the fame and appreciation your journey inside is stopped by another trap. This time it's a fatal trap, you are caught in between two walls and the walls are closing in together. You barely have any time left, before you get squeezed in between the walls and your squad is more dependent on you than ever. This is your supreme test. The only way out is again to crack a riddle, but this time it is not too easy.

Given,

X % A1 = R1
X % A2 = R2
X % A3 = R3
… …. … … ..
X % Ai = R_i

Where $\boldsymbol{A_i}$ is an integer and $\boldsymbol{R_i}$ being the remainder when it divides another integer $\boldsymbol{X}$.
You have to tell the minimum value of $\boldsymbol{X}$ which satisfies this condition.

You are really hungry at this point of time and your friends promise that they will treat you with chinese food if you help them get out. Well, a small prize for saving their life, but your own life depends on it.

INPUT

- The first line contains a single integer *N* denoting the number of integers
- The second line contains *N* spaced integers A_1 A_2 A_3…. A_N where A_i denotes the ith number
- The third line contains *N* spaced spaced integers R_1 R_2 R_3 …. R_N where R_i denotes the i^{th} number.

OUTPUT

- Print a single integer *X* denoting the smallest integer which satisfies all the conditions.

CONSTRAINTS

- 1 <= N <= 10^4

SAMPLE INPUT

```
3
3 4 5
2 3 1
```

SAMPLE OUTPUT

```
11
```

EXPLANATION

When we divide 11 by 3, the remainder is 2
When we divide 11 by 4, the remainder is 3
When we divide 11 by 5, the remainder is 1

Hence 11 satisfies all the conditions

PROBLEM H

Rick Beats Monty

Rick owns a coaching institute, his competitor in profession being Monty. Rick wants to showcase his institute as the best institute but unfortunately his institute is not as good as he portrays and neither are the results. But Rick is smart. He has divided his students into two different batches (groups) and displays as the sum of averages of both the batches. He wants to display a score as high as possible. He seeks your help to find the same. Given a list of ***N*** students and their results (scores) you have to optimally divide them into two groups so as to maximise the sum of averages of both the batches, following which you have to tell the ***maximum sum of averages*** of both the groups possible.

INPUT

- The first line contains a single integer *T* denoting the number of test cases
- The first line of each test case contains contains a single integer *N* denoting the number of children
- The second line of each test case contains *N* spaced integers A_1 A_2 A_3 … A_N denoting the score of each child.

OUTPUT

- Print a single integer denoting the maximum value of the sum of averages of both the groups.

CONSTRAINTS

- 1 <= T <= 100
- 1 <= N <= 10^5
- 1 <= A_i <= 10^5
- Sum of N over all test cases does not exceed 10^5

SAMPLE INPUT

```
1
4
2 2 2 2
```

SAMPLE OUTPUT

```
4
```

EXPLANATION

The numbers can be divided into two groups (2,2) [average = 2] and (2,2) [average = 2] giving the sum of average as 4. This is the maximum possible sum of averages of both the groups