O-Week Dragon Contest 2024 Problems

UNSW CPMSoc

5 February 2024

Contents

Trivial [Maths]

Hello there!

Your first task is to evaluate $2^{10} + 10^3$.

Enter your answer into the box below to submit and check your answer!

For all the tasks, including this one, you can submit as many times as you like with no penalty! Only your best submission will be counted.

Average Squares (Part 1) [Maths]

Isaac the dragon is learning about square numbers, and is looking for nice properties.

Find two distinct positive square numbers whose sum is also a square number.

Submit your answer as a comma-separated list of the two square numbers. For example, if your numbers are 1 and 4, you should submit "1, 4".

Average Squares (Part 2) [Maths]

Isaac the dragon decides to combine what he knows about averages with squares.

Find two distinct positive square numbers whose mean is also a square number.

Submit your answer as a comma-separated list of the two square numbers. For example, if your numbers are 1 and 4, you should submit "1, 4".

Average Squares (Part 3) [Maths]

Isaac the dragon remembers that the average (mean) can also be calculated for more than two numbers at a time.

Find three distinct positive square numbers whose mean is also a square number.

Submit your answer as a comma-separated list of the three square numbers. For example, if your numbers are 1, 4 and 9, you should submit "1, 4, 9".

Years of the Dragon (Part 1) [Maths]

Brian's lucky numbers are 1, 7 and 8, and he wants to see what else these numbers share in common.

Evaluate $\frac{1}{1\times2} + \frac{1}{2\times3} + \frac{1}{3\times4} + ... + \frac{1}{7\times8}$.

Submit your answer as an irreducible fraction, with the numerator and denominator separated by a slash. For example, if your answer is 0.8, you should submit "4/5".

Years of the Dragon (Part 2) [Maths]

Since Brian was born in the Year of the Dragon, he also decides to use previous Years of the Dragon in a similar way as his lucky numbers.

Evaluate $\frac{1}{8\times20}+\frac{1}{20\times32}+\frac{1}{32\times44}+...+\frac{1}{2012\times2024}$.

Submit your answer as an irreducible fraction, with the numerator and denominator separated by a slash. For example, if your answer is 0.8, you should submit "4/5".

Bob's Your Uncle (Part 1) [Maths]

Suppose that in a population of dragons, each dragon:

- has a 10% chance of being named Bob,
- has one father and one mother,
- has 3 sons, and
- has 3 daughters.

If you are a random dragon in this population, what is the probability that Bob's your uncle (that is, the probability that at least one of your parents has at least one brother named Bob)?

Submit your answer as an irreducible fraction, with the numerator and denominator separated by a slash. For example, if your answer is 0.8, you should submit "4/5".

Bob's Your Uncle (Part 2) [Maths]

Suppose that in a population of dragons, each dragon:

- has a 10% chance of being named Bob,
- has one father and one mother,
- has either 0, 1, 2 or 3 sons, each with equal probability, and
- has either 0, 1, 2 or 3 daughters, each with equal probability.

If you are a random dragon in this population, what is the probability that Bob's your uncle (that is, the probability that at least one of your parents has at least one brother named Bob)?

Submit your answer as an irreducible fraction, with the numerator and denominator separated by a slash. For example, if your answer is 0.8, you should submit "4/5".

Functional Equation

Find all functions $f : \mathbb{R} - \{0\} \to \mathbb{R}$ such that

$$f(y + f(x)) = 1 + xf(f(x(y + 1)))$$

for all real x and y satisfying $y + f(x) \neq 0$ and $x(y + 1) \neq 0$.

Submit an answer and an explanation.

Goldilocks and the Three Dragons [Maths]

Once upon a time there were three Dragons, one of them was a Little Wee Dragon, and one was a Median-sized Dragon, and the other was a Great Big Dragon. They each had a bed to sleep in; a little bed for the Little Wee Dragon; and a median-sized bed for the Median-sized Dragon; and a great bed for the Great Big Dragon.

By this time the Three Dragons arrived home, and feeling like an afternoon nap they went upstairs to their bedroom.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Great Big Dragon in his great, rough, gruff voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Little Wee Dragon in his little wee voice.

"SOMEBODY HAS BEEN LYING IN MY BED, AND SHE IS HERE STILL!" said the Median-sized Dragon in his median-sized voice.

Goldilocks has three integers a, b, and c. Find a formula using only min and max which always finds the median of a, b, and c.

Submit a valid formula, where a formula is of the form "a", "b", "c", "min(formula, formula)" or "max(formula, formula)." For example, the following is a valid sequence of formula which always find the minimum of a, b and c.

```
min(
    a,
    min(b, c)
)
```

Note that whitespace is ignored for your convenience.

You will be scored based on how few min and max operations your formula uses. In particular, if you use n min and max operations, you will receive $\max(0, \min(1, (8-n)/4)) \times 100\%$ of the points. In particular, if you use at most 4 min and max operations, you will receive 100% of the points.

Goldilocks and the Five Dragons [Maths]

Once upon a time there were five Dragons, one of them was a Little Wee Dragon, and one was a Rather Small Dragon, and one was a Median-Sized Dragon, and one was a Rather Large Dragon, and the other was a Great Big Dragon. They each had a bed to sleep in; a little bed for the Little Wee Dragon; a rather small bed for the Rather Small Dragon; and a median-sized bed for the Median-Sized Dragon; a rather large bed for the Rather Large Dragon; and a great bed for the Great Big Dragon.

By this time the Three Dragons arrived home, and feeling like an afternoon nap they went upstairs to their bedroom.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Great Big Dragon in his great, rough, gruff voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Little Wee Dragon in his little wee voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Rather Large Dragon in his rather large voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Rather Small Dragon in his rather small voice.

"SOMEBODY HAS BEEN LYING IN MY BED, AND SHE IS HERE STILL!" said the Median-Sized Dragon in his median-sized voice.

Goldilocks has five integers a, b, c, d and e. Find a procedure using only min and max which always finds the median of a, b, c, d and e.

Submit a valid sequence of formulas, where a formula is of the form "a", "b", "c", "d", "e", "min(formula, formula)", "max(formula, formula)", or an integer i which represents the result of a previous formula at position i (1-indexed) in your sequence. The final formula in your sequence should find the median of a, b, c, d and e. For example, the following is a valid sequence of formulas which always find the minimum of a, b and c.

```
min(
    a,
    b
)
min(1, c)
```

Note that whitespace is ignored for your convenience.

You will be scored based on how few min and max operations your formula uses. In particular, if you use n min and max operations, you will receive $\max(0, \min(1, (32 - n)/20)) \times 100\%$ of the points. In particular, if you use at most 12 min and max operations, you will receive 100% of the points.

Goldilocks and the Seven Dragons [Maths]

Once upon a time there were seven Dragons, one of them was a Very Tiny Dragon, and one was a Little Wee Dragon, and one was a Rather Small Dragon, and one was a Median-Sized Dragon, and one was a Rather Large Dragon, and the other was a Great Big Dragon, and one was the Largest Dragon Ever. They each had a bed to sleep in; a very tiny bed for the Very Tiny Dragon; a little bed for the Little Wee Dragon; a rather small bed for the Rather Small Dragon; and a median-sized bed for the Median-Sized Dragon; a rather large bed for the Rather Large Dragon; a great bed for the Great Big Dragon; and the largest bed ever for the Largest Dragon Ever.

By this time the Three Dragons arrived home, and feeling like an afternoon nap they went upstairs to their bedroom.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Great Big Dragon in his great, rough, gruff voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Little Wee Dragon in his little wee voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Rather Large Dragon in his rather large voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Rather Small Dragon in his rather small voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Very Tiny Dragon in his very tiny voice.

"SOMEBODY HAS BEEN LYING IN MY BED!" said the Largest Dragon Ever in his maximally large voice.

"SOMEBODY HAS BEEN LYING IN MY BED, AND SHE IS HERE STILL!" said the Median-Sized Dragon in his median-sized voice.

Goldilocks has five integers a, b, c, d, e, f and g. Find a procedure using only min and max which always finds the median of a, b, c, d, e, f and g.

Submit a valid sequence of formulas, where a formula is of the form "a", "b", "c", "d", "e", "f", "g", "min(formula, formula)", "max(formula, formula)", or an integer i which represents the result of a previous formula at position i (1-indexed) in your sequence. The final formula in your sequence should find the median of a, b, c, d, e, f and g. For example, the following is a valid sequence of formulas which always find the minimum of a, b and c.

```
min(
    a,
    b
)
min(1, c)
```

Note that whitespace is ignored for your convenience.

You will be scored based on how few min and max operations your formula uses. In particular, if you use n min and max operations, you will receive $\max(0, \min(1, (32 - n)/20)) \times 100\%$ of the points. In particular, if you use at most 12 min and max operations, you will receive 100% of the points.

Arbitrage-Free [Maths]

Let $S = \{1, 2, ..., 2024\}$.

We say a function $c : S \to \mathbb{R}$ is *arbitrage-free* if, for all functions $w : S \to \mathbb{R}$, there exists some $k \in \mathbb{Z}$ such that

$$\sum_{i \in S} w(i)(\max(0, k - i) - c(i)) \leq 0.$$

Show that c is arbitrage-free if and only if it is:

- non-negative (that is, $c(x) \geq 0$ for all $x \in S$),
- non-increasing, but not too non-increasing (that is, $0 \leq c(x) - c(x + 1) \leq 1$ for all $x \in S \setminus \{2024\}$), and
- convex (that is, $c(x - 1) + c(x + 1) \geq 2c(x)$ for all $x \in S \setminus \{1, 2024\}$).

Year of the Dragon [Programming]

Program time limit: 1 second
Program memory limit: 512 MB

Did you know that 2024 is the year of the dragon?

In fact, any year which is 8 more than a multiple of 12 is the year of the dragon.

Given a positive integer N, determine whether it is the year of the dragon.

Input

On the first and only line of input, you will be given one integer N.

You should read from standard input. In Python, you could use the line

```
N = int(input())
```

. In C or C++, you could use the line:

```
int N; scanf("%d", &N);
```

.

Constraints

Each input case will satisfy the following constraints:

- $1 \leq N \leq 10\,000$

Output

You should output "YES" if N is the year of the dragon, and "NO" otherwise.

You should write to standard output. In C or C++, you could use a line like `printf("YES\n");`. In Python, you could use a line like `print("YES")`.

Sample Input 1

```
2024
```

Sample Output 1

```
YES
```

Explanation 1

2024 is the year of the dragon, since 2024 is 8 more than 2016, which is a multiple of 12.

Sample Input 2

```
2023
```

Sample Output 2

```
NO
```

Explanation 2

2023 is not the year of the dragon.

Scoring

Your program will be tested on 5 inputs one after another, 2 of which are sample inputs above, and 3 of which are secret inputs. Your score for the submission will be proportional to the amount of test cases for which you output the correct answer.

Tetris [Programming]

Program time limit: 1 second
Program memory limit: 512 MB

You are playing a modified version of Tetris where the board is width W, where W is either 2 or 3. There are N rounds, where in each round, you must drop an L-shaped piece, which you may flip, rotate or shift however you like. What is the maximum number of rows you can fill?

Input

On the first and only line of input, you will be given two integers N and W.

You should read from standard input. In Python, you could use the line

```
N, W = map(int, input().split())
```

. In C or C++, you could use the line:

```
int N, W; scanf("%d%d", &N, &W);
```

.

Constraints

Each input case will satisfy the following constraints:

- $1 \le N \le 1\,000$
- W is either 2 or 3

For 40% of the test cases, $W = 2$, and for the other 60% of the test cases, $W = 3$.

Output

Your output should be a single integer, the maximum possible number of rows you can clear.

You should write to standard output. In C or C++, you could use the line `printf("%d", answer);`. In Python, you could use the line `print(answer)`.

Sample Input 1

```
3 2
```

Sample Output 1

```
5
```

Explanation 1

In this case, the board has width 2 and we have 3 pieces, which we could place as follows:

```
 c
 c
cc
bb
ab
ab
aa
```

Here, 5 rows are filled, which is the maximum possible.

Sample Input 2

```
2 3
```

Sample Output 2

```
2
```

Explanation 2

In this case, the board has width 3 and we have 2 pieces, which we could place as follows:

```
bbb
a b
aaa
```

Here, 2 rows are filled, which is the maximum possible.

Scoring

Your program will be tested on 10 inputs one after another, 2 of which are sample inputs above, 3 of which are secret inputs with $W = 2$, and 5 of which are secret inputs with $W = 3$. Your score for the submission will be proportional to the amount of test cases for which you output the correct answer.

Hestian Triples [Programming]

Program time limit: 1 second
Program memory limit: 512 MB

Hestia gazes into the night sky and tries to count the stars. Of the constellations she finds her favourite: Serpens, Hydra, Draco, but altogether there are too many stars to count. Despite their abundance, she thinks that counting the triples of an array would be faster.

A *Hestian triple* is a triplet of integers i, j and k such that $1 \le i < j < k \le n$ and $a_i \times a_j = a_k$. She wonders, given an array a, how many Hestian triples does it contain?

Help Hestia find the answer.

Input

The first line of input contains one integer n, the length of the array a.
The next line has n whitespaced integers, $a_1, a_2, \cdots a_n$ where a_i is the ith element of a.
You should read from standard input.

Constraints

Each input case will satisfy the following constraints:

- $1 \le n \le 10\,000$
- for all $1 \le i \le n$, $1 \le a_i \le 10^{18}$
- for all $1 \le i < j \le n$, $1 \le a_i \times a_j \le 10^{18}$

Output

Output one integer, the number of Hestian Triples in a.
You should write to standard output.

Sample Input 1

```
6
5 2 3 10 2 6
```

Sample Output 1

```
3
```

Sample Input 2

```
5
5 4 4 20 15
```

Sample Output 2

```
2
```

Explanation 1

There are exactly 3 Hestian Triples
$a_1 \times a_2 = a_4$ $(5 \times 2 = 10)$
$a_2 \times a_3 = a_6$ $(2 \times 3 = 6)$
$a_3 \times a_5 = a_6$ $(3 \times 2 = 6)$

Explanation 2

There are exactly 2 Hestian Triples
$a_1 \times a_2 = a_4$ $(5 \times 4 = 20)$
$a_1 \times a_3 = a_4$ $(5 \times 4 = 20)$

Scoring

Your program will be tested on multiple inputs one after another. Your score for the submission will be 100% if your program outputs the correct answer for all of these inputs.

Colourful Roads [Programming]

Program time limit: 1 second
Program memory limit: 512 MB

In Dragonia, there are N towns, and M one-way roads between towns, where each road has a colour. For a dragon to walk along a road, they must be wearing a coloured hat corresponding to the colour of that road. What is the fewest times a dragon needs to change their hat to get from house S to house E?

Input

On the first line of input, you will be given two integers N and M, denoting the number of towns and edges respectively. On the next line, you will be given two integers S and E, denoting the starting and ending town. The following M lines will each contain 3 numbers, x_i, y_i, and c_i, representing a directed road with colour c_i from town x_i to town y_i.

You should read from standard input.

Constraints

Each input case will satisfy the following constraints:

- $1 \leq N \leq 100\,000$
- $1 \leq M \leq 200\,000$
- $1 \leq S, E \leq N$
- $1 \leq x_i, y_i \leq N$
- $1 \leq c_i \leq 100\,000$

Subtask 1: For 20% of the points, $c_i = 1$ for all i.
Subtask 2: For 30% of points, $c_i = 1$ or $c_i = 2$ for all i and you are guaranteed that for any valid path, the colours will alternate between 1 and 2.
Subtask 3: For the final 50% of points, there are no additional constraints.

Output

If there is no path from S to E, output "IMPOSSIBLE". Otherwise, output the minimum number of hat changes you have to make.

You should write to standard output.

Sample Input 1

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

Sample Output 1

```
1
```

Explanation 1

To reach town 5 from town 1 with the minimum amount of hat changes, we can elect to start with a hat of colour 1 then follow the route $1 \to 4 \to 5$.
Note that this input also conforms to the requirements of subtask 2.

Scoring

For each subtask (worth 20%, 30% and 50%; see the Constraints section), your program will be tested on multiple secret input cases one after another. You will recieve points for a subtask if your program passes **all** the test cases for that subtask. Your score is the sum of the points from all the subtasks your submission passes. Recall that your final score on the task is the score of your highest scoring submission.

Mod Thing [Programming]

Program time limit: 6 seconds
Program memory limit: 512 MB

Dr. Agon wants your help.

He gives you non-negative integers $a_1, a_2, \cdots, a_N$. He allows you to choose any integer M, ($M \geq 2$), and wants you to minimise $\sum_{i=1}^{N}(a_i \bmod M)$.

Find the minimum of this sum.

Input

On the first line of input, you will be given an integer N, denoting the number of integers you will receive.
On the next line, you will be given the N integers a_1 up to a_N.
You should read from standard input.

Constraints

Each input case will satisfy the following constraints:

- $1 \leq N \leq 100\,000$
- $0 \leq a_i \leq 100\,000$

Subtask 1: For 50% of the points, $N \leq 1000$ and $a_i \leq 1000$.
Subtask 2: For the final 50% of points, there are no additional constraints.

Output

Your output should contain one number: the minimum sum of mods by choosing M optimally.
You should write to standard output.

In C or C++, please use the long long type instead of int type to store numbers, because they may be big.

Sample Input 1

```
7
15 25 16 95 42 31 35
```

Sample Output 1

```
4
```

Explanation 1

Choosing $M = 5$, we have $(15\%5) + (25\%5) + (16\%5) + (95\%5) + (42\%5) + (31\%5) + (35\%5) = 4$.
It can be shown that for all other $M > 1$, the value will be more than 4.

Scoring

For each subtask (worth 50%, and 50%; see the Constraints section), your program will be tested on multiple secret input cases one after another. You will recieve points for a subtask if your program passes **all** the test cases for that subtask. Your score is the sum of the points from all the subtasks your submission passes. Recall that your final score on the task is the score of your highest scoring submission.

Ianuarian Triples [Programming]

Program time limit: 1 second
Program memory limit: 512 MB

Ianus sees Hestia counting stars and wonders if he can create an array of his own that mirrors her favourite constellations. He knows what size arrays Hestia likes and the number of stars in Hydrus, but he isn't certain if it is even possible to marry these two numbers together.

A *Ianuarian triple* is a triplet of integers i, j and k such that $1 \leq i < j < k \leq n$ and $a_i \times a_j = a_k$. He wonders, does there exist an array a of length n, **composed of only 2s and 4s**, which has exactly m Ianuarian triples? If so, help Ianus construct such an array.

Input

The first line contains one integer t, the number of testcases.
The next t lines contain two integers n and m, the target length of the array a and the target number of Ianuarian Triples.
You should read from standard output.

Constraints

- $1 \leq t \leq 10\,000$
- $1 \leq n \leq 10\,000$
- $1 \leq m \leq 10\,000$

Output

If there does not exist an array of length n with m Ianuarian Triples, output "NO".
If there exists such an array output "YES". On the next line output the array, a string of length n consisting of characters '2' and '4'.
You should write to standard output.

Sample Input 1

```
3
6 2
7 7
10 100
```

Sample Output 1

```
YES
422442
YES
2424244
NO
```

Explanation 1

Testcase 1
422442 has length 6 and has exactly 2 Ianuarian Triples:
$a_2 \times a_3 = a_4$
$a_2 \times a_3 = a_5$

Testcase 2
2424244 has length 7 and has exactly 7 Ianuarian Triples:

$a_1 \times a_3 = a_4$
$a_1 \times a_3 = a_6$
$a_1 \times a_3 = a_7$
$a_1 \times a_5 = a_6$
$a_1 \times a_5 = a_7$
$a_3 \times a_5 = a_6$
$a_3 \times a_5 = a_7$

Testcase 3
It can be shown that there exists no array of length 10 with exactly 100 Ianuarian Triples.

Scoring

Your program will be tested on multiple inputs one after another. Your score for the submission will be 100% if your program outputs the correct answer for all of these inputs.

Secret Stall Questions [In-Person]

Did you solve our in-person o-week stall questions? Combine your secret words to solve the problem.