2025 Rocket Code Rumble Problems

UNSW CPMSoc & Rocketry

14 November 2025

Contents

Six Seven [Maths]

Our problemsetters are having an argument!!!

Oscar claims that 6^7 is larger than 7^6 while Josh thinks 7^6 is larger. Frank thinks they are the same size.

Who is correct? Enter their name with proper capitalisation.

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.

Rocket Economics [Maths]

A passenger rocket has 4 modular sections, which can each either be economy class (fitting 7 passengers), business class (fitting 4 passengers), or first class (fitting just 1 passenger). How many different passenger capacities does the rocket have?

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.

Rocket Range [Maths]

A rocket with three fuel tanks has a range of $49 + 42 + 36 = 127$ km. If each additional fuel tank increases the range by 6/7 of the previous increase, then what is the range of a rocket with five fuel tanks, in km?

Submit your answer as an irreducible fraction, with the numerator and denominator separated by a slash. For example, if your answer is 1.8, you should submit "9/5".

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.

Rocket Trajectory [Maths]

A rocket follows the arc of a concave down parabola as it flies. It has a constant horizontal velocity. At time $t = 4$ minutes after launch, the rocket has altitude $H = 65$ km. Similarly, for $t = 6$ minutes, $H = 65$ km, and for $t = 7$ minutes, $H = 59$ km. Find the altitude, in km, for $t = 10$ minutes.

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.

Grid [Maths]

How many ways are there to paint the cells of an $N \times N$ grid black and white such that every row and every column has exactly 2 black cells?

Subtask 1 (30% of points)

$N = 2$

Subtask 2 (30% of points)

$N = 3$

Subtask 3 (40% of points)

$N = 4$

Submission

Submit your answer to the subtasks as a comma-separated list of integers. For example, if your answers to the subtasks are 6, 7 and 67, you should submit `6, 7, 67`. Note that if you have not solved a subtask, you can submit a dummy answer for that subtask. For example, if your answer to the first subtask is 3, you could submit `3,0,0`.

Engine Symmetries [Maths]

The propulsion system on a rocket consists of three circular layers of engines: an outer layer with 9 engines, a middle layer with 7 engines, and an inner layer with 5 engines. Each engine may use solid, liquid or hybrid propellant. When observed from the bottom, how many distinct engine configurations exist? Note that if one configuration can be changed into another just by rotating any of the layers, they are considered to be the same configuration.

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.

Burnout [Maths]

A hybrid rocket engine consists of a hybrid propellant: solid fuel grain and liquid oxidizer. The solid fuel grain takes the shape of a cylinder with a cylindrical hole through the middle called the port. The grain is then ignited and liquid oxidizer is injected into the port hole, causing the fuel to burn/regress outward radially, where the burn rate r, in m/s, is given by

$$r = a\left(\frac{\dot{m}_{\text{ox}}}{A_p}\right)^n$$

,

where $\dot{m}_{\text{ox}}$ denotes the oxidizer injection mass flow rate and A_p is the port area, which is dependant on the port radius.

Take $a = 2.6676 \times 10^{-5}$, $n = 0.72493$ and suppose that the oxidizer injection mass flow rate is a constant 1 kg/s.

Given that the entire fuel grain has an outer radius of 0.034 m and the port radius is initially 0.0175 m, determine how long the hybrid engine will fire before burning through the entire fuel grain.

Submit your answer rounded to the nearest second. For example, if your answer is 3.2, you should submit "3".

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.

Lightning Fast Rocket [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

Naman has arrived at the **Commercial Planetary Market (CPM)** to buy a rocket.
As a fan of speed, he wants the rocket with the **greatest potential speed**.

Each rocket is made by a different company and described by:

- model name
- exhaust velocity
- initial mass
- final mass

Naman gives you the formula to calculate a rocket's potential speed:

$$V_{\text{potential}} = V_{\text{exhaust}} \times \ln\left(\frac{m_{\text{initial}}}{m_{\text{final}}}\right)$$

Your task is to help Naman to determine which rocket has the greatest potential speed.

Input

- The first line contains an integer N — the number of rockets.
- The next N lines each contain a string S_i and three integers A_i, B_i, and C_i, representing respectively:
  - S_i: model name
  - A_i: exhaust velocity
  - B_i: initial mass
  - C_i: final mass

Constraints

For all test cases:

- $2 \le N \le 10^5$
- $1 \le A_i, B_i, C_i \le 10^9$
- No two rockets share the same name.
- $B_i > C_i$

Output

- Print the name of the rocket with the greatest potential speed.
- If multiple rockets have the same potential speed, output the one with the **lexicographically smallest** name.

Templates

You should read from standard input and write to standard output.

In Python, you could use the following code.

```python
# Taking inputs, already done! :D
N = int(input())
names = []
```

```python
A = []
B = []
C = []
for i in range(N):
    name, a, b, c = input().split()
    names.append(name)
    A.append(int(a))
    B.append(int(b))
    C.append(int(c))

# Write your code here


# Printing output, currently just prints the first name given
print(names[0])
```

In C or C++, you could use the following code.

```c
// Taking inputs, already done! :D
int N; scanf("%d", &N);
string names[N];
int A[N], B[N], C[N];
for (int i = 0; i < N; i++) scanf("%s %d %d %d", &names[i], &A[i], &B[i], &C[i]);

// Write your code here


// Printing output, currently just prints the first name given
printf("%s", names[0])
```

Sample Input

```
2
Saturn_V 2580 2970000 40100
Space_Shuttle 2770 549052 27200
```

Sample Output

```
Saturn_V
```

Explanation

For each rocket:

- **Saturn V** $\rightarrow 2580 \times \ln(2970000/40100) \approx 11106.75$
- **Space Shuttle** $\rightarrow 2770 \times \ln(549052/27200) \approx 8323.78$

Since $11106.75 > 8323.78$, **Saturn V** is the correct answer.

Scoring

Your program will be tested on the sample case and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.

Countdown [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

CPMsoc and Rocketry worked on a rocket together. However, as the rocket was about the launch, the person doing the countdown fell asleep on their job, causing one number to not be counted. The proper countdown should start with n, and decrement by 1 until we reach 1. Help us find out which number was missing from the countdown!

Input

- The first line of input contains one integer, n, which is the length of the proper countdown.
- The second line of input contains $n - 1$ space separated integers, which will all be unique integers in the range $[1 \dots n]$, and one of the integer in this sequence will be missing, hence only n-1 integers.

Constraints

For all test cases:

- $1 \leq n \leq 100000$.

Output

- Output a single integer, the countdown number that went missing.

Templates

You should read from standard input and write to standard output.

In Python, you could use the following code.

```python
# Taking inputs, already done! :D
N = int(input())
counts = list(map(int, input().split()))

ans = N
# Write your code here


# Printing output
print(ans)
```

In C or C++, you could use the following code.

```c
// Taking inputs, already done! :D
int N; scanf("%d", &N);
int W[N-1]; for (int i = 0; i < N-1; i++) scanf("%d", &W[i]);

int ans;
// Write your code here


// Printing output
printf("%d", &ans);
```

Sample Input 1

```
6
6 5 3 2 1
```

Sample Output 1

```
4
```

Explanation 1

In this case, the proper count down should be `6 5 4 3 2 1`, but only `6 5 3 2 1` is given, so the missing countdown number is 4.

Sample Input 2

```
10
10 9 8 6 5 4 3 2 1
```

Sample Output 2

```
7
```

Explanation 2

In this case, the proper count down should be `10 9 8 7 6 5 4 3 2 1`, but only `10 9 8 6 5 4 3 2 1` is given, so the missing countdown number is 7.

Scoring

Your program will be tested on the 2 sample cases and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.

Arugula [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

Anna has a bag of N arugula leaves, and each leaf has a weight in milligrams, w_i. She wants to split the leaves into two salads so she can share with her friend Dom. However, Dom makes the request that he wants the weight of the salads to have the same parity. Two numbers have the same parity if they are both even, or if they are both odd.

Output "yes" if Dom's request is possible, otherwise output "no". Importantly, the two salads do not need to contain the same number of leaves. However, each salad must contain at least one leaf.

Input

- The first line of input contains an integer N, which represents the total number of leaves,
- The second line contains N space separated integers, representing the weights of the leaves.

Constraints

For all test cases:

- $1 \leq N \leq 200,000$,
- $1 \leq w_i \leq 10,000$ for $1 \leq i \leq N$.

Output

- Output either "Yes" or "No", to say if Dom's request is possible.

Templates

You should read from standard input and write to standard output.

In Python, you could use the following code.

```python
# Taking inputs, already done! :D
N = int(input())
W = list(map(int, input().split()))

flag = False
# Write your code here


# Printing output
if flag == True:
    print("Yes")
else:
    print("No")
```

In C or C++, you could use the following code.

```c
// Taking inputs, already done! :D
int N; scanf("%d", &N);
int W[N]; for (int i = 0; i < N; i++) scanf("%d", &W[i]);

bool flag;
// Write your code here
```

```
// Printing output
if (flag == true) {
    printf("Yes\n");
} else {
    printf("No\n");
}
```

Sample Input 1

```
4
10 6 7 3
```

Sample Output 1

```
Yes
```

Explanation 1

One possible way to make the salads is for Anna to take just the first leaf, and for Dom to have the other three. In this case, the weights are 10 and $6 + 7 + 3 = 16$. Since both weights are even, this satisfies Dom's request.

Sample Input 2

```
3
2 4 1
```

Sample Output 2

```
No
```

Explanation 2

It can be shown that no matter how the leaves are divided, one of them will have an odd weight, and one will have an even weight. Therefore, it is impossible to satisfy Dom's request.

Scoring

Your program will be run on both sample cases and 9 secret cases one after another, and if it produces the correct output for **all** test cases, it solves this task. Recall that your final score on the task is the score of your highest scoring submission.

Moon Art I [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

Esther has safely landed on the Moon. As a huge fan of the numbers 6 and 7, she wants to draw a giant 67 on the lunar surface. What's better than a 67 you ask? It's a bigger 67!

However, Esther didn't have anything suitable for drawing, she dismantles her Lunar Roving Vehicle and obtains N identical metal rods. Each rod is either placed horizontally, which is represented by an underscore `_`, or placed vertically, which is represented by a vertical bar `|`. With these rods, she will construct the digits 6 and 7 next to each other in a fixed style, but as big as possible.

Here are the examples for 67 s which require 9, 18, and 27 rods respectively.

```
 _  _
|_   |
|_|  |
```

```
 _ _  _ _
|        |
|_ _     |
|   |    |
|_ _|    |
```

```
 _ _ _  _ _ _
|            |
|            |
|_ _ _       |
|     |      |
|     |      |
|_ _ _|      |
```

Input

- The first and only line contains an integer N — the number of metal rods.

You should read from standard input.

Constraints

For all test cases:

- $9 \le N \le 1000$

Output

- Print the drawing of the biggest 67 you can make with the given metal rods. You do not need to use all the metal rods.
- Note that, you **should not** print any extra leading or tailing empty spaces.

You should write to standard output.

Sample Input

```
10
```

Sample Output

```
 _  _
|_   |
|_|  |
```

Explanation

With 10 metal rods, that is enough to construct the smallest version of 67, but not the next largest version, which requires 18 rods.

Scoring

Your program will be tested on the sample case and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.

Moon Art II [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

Susan has also safely landed on the Moon. There, she saw a huge "67" built using metal rods. Since Susan didn't like these two numbers, she decided to take it apart. Using the collected N metal rods, she decided to build a road instead.

However, she didn't want to build just any road — she wanted each tile of the road to have the shape of a Tetris piece. Since she liked all the pieces equally, she wanted to make sure that the number of occurences of any two shapes did not differ by more than 1.

What is the maximum length of the road that Susan can build following these rules?

Here are the Tetris shapes built using the metal rods:

```
 _ _
|   |
|_ _|
 _ _ _ _
|_ _ _ _|
   _ _
 _|  _|
|_ _|
 _ _
|_  |_
  |_ _|
 _ _ _
|  _ _|
|_|
 _ _ _
|_ _  |
    |_|
 _ _ _
|_   _|
  |_|
```

Input

- You are given the picture of numbers 67 on some number of lines.

You should read from standard input.

Constraints

- $9 \leq N \leq 1000$

Output

- Print the length of the longest path, Susan can build.

You should write to standard output.

Sample Input 1

```
 _  _
|_   |
|_|  |
```

Sample Output 1

```
2
```

Explanation 1

You have 9 rods, which you can use to make the 2×2 square piece, which is 2 square wide.

Sample Input 2

```
 _ _  _ _
|        |
|_ _     |
|   |    |
|_ _|    |
```

Sample Output 2

```
6
```

Explanation 2

You have 18 rods, which you can use to make the 1×4 tetris piece and 2×2 square piece, which is $4 + 2 = 6$ square wide.

Scoring

Your program will be tested on the sample case and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.

Moon Art III [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

Yoshi has also safely landed on the Moon, where she found the playful Labubus jumping around on a road with a Tetris-like structure.
After observing them for a while, she discovered that the Labubu on the i-th tile always teleports to the T_i-th tile.

Now Yoshi is wondering: if a Labubu starts on the A_i-th tile and makes B_i teleportations, where will it end up?
To be sure, Yoshi wants to answer this question M times.

Yoshi has also safely landed on the Moon, where she found the playful Labubus playing on a road that has Tetris like structure. After observing the Labubus for a while, she discovered that the Labubus on the i-th tile always teleports to the T_i-th tile.

Now Yoshi is wondering: if a Labubu starts on the A_i-th tile and makes B_i teleportations, where will the Labubu end up? Just to make sure she understands the habit of Labubus, Yosh wants to answer this question M times.

Input

- The first line of input contains an integer N — the number of tiles the road has.
- In the next line, there are N numbers $T_1, T_2, \ldots, T_N$ - the position Labubu has teleported to from i-th tile.
- The next line of input contains an integer M - the number of queries to answer.
- In the following M lines, there are two integers A_i and B_i - the starting location, and the number of teleportations.

You should read from standard input.

Constraints

- $1 \le N, M \le 200'000$
- $1 \le A_i, T_i \le N$
- $1 \le B_i \le 1'000'000'000$

Output

- For each query, print the position, the Labubu would end up on.

You should write to standard output.

Sample Input

```
10
2 3 4 5 6 7 8 9 10 1
3
1 4
5 2
10 3
```

Sample Output

```
5
7
```

```
3
```

Explanation

Labubu started on tile 1 would move to $1 - 2 - 3 - 4 - 5$.
Labubu started on tile 5 would move to $5 - 6 - 7$.
Labubu started on tile 10 would move to $10 - 1 - 2 - 3$.

Scoring

Your program will be tested on the sample case and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.

KFC Rocket [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

Space Colonel Natalie is trying a new high-efficiency way to fry chicken - using rocket exhaust!

She places N chicken pieces in a line, each with a spiciness level S_i.

Every time a rocket launches, the exhaust of that rocket passes over a range of chickens from L_i to R_i, increasing their spiciness by a A_i. The Colonel Natalie wants to see how spicy they've become in total. However, if a chicken is spicier than K, it's going to become burnt, and will lose it's spiciness forever.

You, as the official Rocket Chicken Engineer, need to keep answer Colonel Natalie's question.

Input

- The first line of input contains an integer N, M, K — the number of chicken pieces in a line, the number of rockets, and the maximum spiciness value of a chicken piece.
- In the next line, there are N numbers $S_1, S_2, \ldots, S_N$ - the initial spiciness of the chicken pieces.
- In the following M lines, there are three integers L_i, R_i and A_i - the left and rightmost chicken pieces affected, and the increase in spiciness.

You should read from standard input.

Constraints

- $1 \le N, M \le 10^6$
- $1 \le K \le 10^9$
- $1 \le L_i, R_i \le N$
- $1 \le A_i, S_i \le 10^9$
- $S_i \le K$

Output

- Print the total spiciness of all chicken pieces after every rockets are launched.

You should write to standard output.

Sample Input

```
6 3 5
1 1 2 2 3 3
1 2 1
2 3 1
4 6 3
```

Sample Output

```
12
```

Explanation

Final state is: `2 3 3 5 6 6`, here the 5-th and 6-th chicken pieces are burnt.

Scoring

Your program will be tested on the sample case and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.

SixSeven [Programming]

Program time limit: 1 second

Program memory limit: 512 MB

You have two numbers a and b, which are initially 6 and 7.

You must perform one of four operations Q times:

- `6`: Add the digit 6 to the start of a
- `7`: Add the digit 7 to the start of b
- `a x`: Change the x-th digit of a (from the left, 1-indexed). A 6 becomes a 7 and vice versa.
- `b x`: Change the x-th digit of b (from the left, 1-indexed). A 6 becomes a 7 and vice versa.

After each operation, output which of a and b is larger or if they are equal.

Input

- The first line contains an integer Q — the number of operations.
- The next Q lines each contain an operation. Each operation is one of:
  - `6` — add the digit 6 to the start of a
  - `7` — add the digit 7 to the start of b
  - `a x` — change the x-th digit of a (where x is an integer)
  - `b x` — change the x-th digit of b (where x is an integer)

You should read from standard input.

Constraints

- $1 \le Q \le 10^6$
- For operations `a x` and `b x`: $1 \le x \le$ (current number of digits in the number)

Output

After each operation, output a single line:

- `a` if $a > b$
- `b` if $b > a$
- `e` if $a = b$

You should write to standard output.

Sample Input

```
5
6
7
a 1
a 2
b 2
```

Sample Output

```
a
b
b
e
a
```

Explanation

After each operation, we compare the current values of a and b:

- After `6`: $a = 66$, $b = 7$, output a.
- After `7`: $a = 66$, $b = 77$, output b.
- After `a 1`: $a = 76$, $b = 77$, output b.
- After `a 2`: $a = 77$, $b = 77$, output e.
- After `b 2`: $a = 77$, $b = 76$, output a.

Scoring

Your program will be tested on the sample case and several hidden cases.
If it produces the correct output for **all** test cases, it will be considered correct.
Recall that your final score for this problem is the score of your highest-scoring submission.

Moon Escape [Programming]

Program time limit: 2 second

Program memory limit: 512 MB

The Legendary Hero of Labubus Noah has noticed that the mischievous Labubu Jerry has escaped thanks to the huge traffic on the moon. But thatnks to the Commercial Planetary Market (CPM) moon branch he have access to the logs of people who have been on the moon.

The logs indicate that a person has been on the moon, at station S_i on days A_i to B_i ($A_i \leq B_i$).

Noah have done some investigation, and he narrowed down the day Jerry's escape happened, to M different days. According to the investigation, the escape could have happened on day D_i, between stations C_i and D_i. More specifically, for all of the M days: $D_1, D_2, \ldots, D_M$, he has to check everyone's logs who were in station S which is in range (C_i, D_i).

Then, for all of the M days, how many logs, does Noah have to check to find Jerry?

The legendary hero of the Labubus, Noah, has discovered that the mischievous Labubu Jerry has escaped, taking advantage of the heavy traffic on the Moon.
Fortunately, thanks to the Commercial Planetary Market (CPM) Moon branch, Noah has access to the visitor logs of everyone who has been on the Moon.

Each log entry states that a person was at station S_i from day A_i to day B_i (where $A_i \leq B_i$).

After some investigation, Noah has narrowed down the possible days of Jerry's escape to M specific days. For each of these days D_i, Noah suspects that the escape could have occurred between stations L_i and R_i. In other words, for every day D_i, Noah must check all logs of people who were at any station S such that $L_i \leq S \leq R_i$ **on day D_i**.

For each of the M days, determine how many logs Noah needs to check to possibly find Jerry.

Input

- The first line contains two integers N and M — the number of logs and the number of days in question.
- The next N lines each contain three integers A_i, B_i, and S_i — indicating that a person was at station S_i from day A_i to day B_i (inclusive).
- The next M lines each contain three integers L_i, R_i, and D_i — representing that Jerry's escape could have happened on day D_i between stations L_i and R_i.

Constraints

- $1 \leq N, M \leq 5 \cdot 10^5$
- $1 \leq S_i \leq 5 \cdot 10^5$
- $1 \leq D_i \leq 5 \cdot 10^5$
- $1 \leq A_i \leq B_i \leq 5 \cdot 10^5$
- $1 \leq L_i \leq R_i \leq 5 \cdot 10^5$

Output

- For each of the M days, print a single integer — the number of logs that match the given day and station range.

You should write to standard output.

Sample Input

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

Sample Output

```
2
3
```

Explanation

- On day 3, people 1 and 2 were at stations within $[1, 2]$.
- On day 2, people 1, 2, and 3 were at stations within $[1, 2, 3, 4]$.

Scoring

Your program will be tested on the sample case and several hidden cases. If it produces the correct output for **all** test cases, it will be considered correct. Recall that your final score for this problem is the score of your highest-scoring submission.