September 14, 2025

DOUBT SESSION

Ques 1. Prime Number

https://codeforces.com/problemset/problem/359/C

Simon has a prime number x and an array of non-negative integers $a_1, a_2, \ldots, a_n$.

Simon loves fractions very much. Today he wrote out number $\frac{1}{x^{a_1}} + \frac{1}{x^{a_2}} + \ldots + \frac{1}{x^{a_n}}$ on a piece of paper. After Simon led all fractions to a common denominator and summed them up, he got a fraction: $\frac{s}{t}$, where number t equals $x^{a_1 + a_2 + \ldots + a_n}$. Now Simon wants to reduce the resulting fraction.

Help him, find the greatest common divisor of numbers s and t. As GCD can be rather large, print it as a remainder after dividing it by number 1000000007 ($10^9 + 7$).

The expression is:

$$\frac{1}{x^{a_1}} + \frac{1}{x^{a_2}} + \cdots + \frac{1}{x^{a_n}} = \frac{s}{t}$$

Let us assume $p = (a_1 + a_2 + a_3 + \cdots + a_n)$

$$\rightarrow \frac{s}{t} = \frac{x^{p-a_1} + x^{p-a_2} + x^{p-a_3} + \cdots x^{p-a_n}}{x^p}$$

Have to find $GCD\left(\sum_{i=1}^{n} x^{p-a_i}, x^p\right)$

$\Rightarrow$ ans will be of the form $x^{\overset{\text{some exponent}}{-}}$

Have to find largest value of ans such that $\left(\sum x^{p-a_i}\right)$ is divisible by x^{-}

Example: $x = 2$

$$\begin{aligned}
S &= 2^3 + 2^4 + 2^8 + 2^{11} + 2^3 \\
&= 2^3(1 + 2^1 + 2^5 + 2^8 + 1) \\
&= 2^3 (2 + 2 + 2^5 + 2^8) \\
&= 2^3 \cdot 2(1 + 1 + 2^4 + 2^7) \\
&= \underbrace{(2^3 \cdot 2 \cdot 2)}(1 + 2^3 + 2^6)
\end{aligned}$$

From here we derive that the largest power that will divide the expression is 5.

$GCD = 2^5$

→ If some x^a is present x times, then you can merge it to get x^{a+1}.

Algo for this problem:

```
map<power, freq>
Find the smallest power;
Count the frequency
if (freq % x == 0)
{  // able to merge to get high pow
   mp[power + 1] += freq/x;
   delete power from map
}
else
   return power
```

Ques 2. Find (A.B)%M where A, B, M <= 10^{18} and only use long long data type not int_128

Suppose we have

int a = 1e6, b = 1e6

if we do (a*b)%(10^9+7)

That is incorrect since

int * int ⇒ int

1e6 * 1e6 ⇒ 1e12 (Overflow)

So,

Binary Form:

$B = 2^{a_1} + 2^{a_2} + 2^{a_3} + \cdots + 2^{a_n}$

AlgoZenith

$A*B = A*2^{a_1} + A*2^{a_2} + --- + A*2^{a_n}$

Taking mod on both sides, m

$\Rightarrow ((A*2^{a_1})\%m + (A*2^{a_2})\%m + --- + (A*2^{a_n})\%m)\%m$

The expression $A*2^{a_1}$ implies:

if 2^{a_1} is 4 $\Rightarrow \underbrace{(A+A+A+A)}_{4 \text{ times}}$

So,

$A*4 \Rightarrow (((A*A)\%m+A)\%m+A)\%m$

$A*8 \Rightarrow (A++---A)$ 8 times

Use Binary Exponentiation

```
while(b>0)
{
    if(b&1)
        ans=(ans+a)%m;
    a=(a+a)%m;
    b/=2;
}
```

$\Rightarrow O(\log B)$

Ques 3. Yet Another Broken Keyboard

https://codeforces.com/problemset/problem/1272/C

Recently, Norge found a string $s = s_1 s_2 \ldots s_n$ consisting of n lowercase Latin letters. As an exercise to improve his typing speed, he decided to type all substrings of the string s. Yes, all $\frac{n(n+1)}{2}$ of them!

A substring of s is a non-empty string $x = s[a \ldots b] = s_a s_{a+1} \ldots s_b$ $(1 \le a \le b \le n)$. For example, "auto" and "ton" are substrings of "automaton".

Shortly after the start of the exercise, Norge realized that his keyboard was broken, namely, he could use only k Latin letters $c_1, c_2, \ldots, c_k$ out of 26.

After that, Norge became interested in how many substrings of the string s he could still type using his broken keyboard. Help him to find this number.

Given some string,

S= abacaba

letters = a, b (valid, allowed)

Example:

→ a b a c a b a

– 1 1 1 0 1 1 1 — use 1 for a, b else 0

For any substring to be valid, it should be continuous 1.

No. of substring will be $= 2\left(\frac{3*4}{2}\right)$

$= 12$

→ s a d f a a s d d a

0 1 1 1 1 1 0 1 1 1

$\frac{5 \times 6}{2} = 15 \quad + \quad \frac{3 \times 4}{2} = 6 \quad = 21$

Ques 4. K-th Beautiful String

https://codeforces.com/problemset/problem/1328/B

For the given integer n $(n > 2)$ let's write down all the strings of length n which contain $n-2$ letters 'a' and two letters 'b' in **lexicographical** (alphabetical) order.

Recall that the string s of length n is lexicographically less than string t of length n, if there exists such i $(1 \le i \le n)$, that $s_i < t_i$, and for any j $(1 \le j < i)$ $s_j = t_j$. The lexicographic comparison of strings is implemented by the operator < in modern programming languages.

For example, if $n = 5$ the strings are (the order does matter):

1. aaabb
2. aabab
3. aabba
4. abaab
5. ababa
6. abbaa
7. baaab
8. baaba
9. babaa
10. bbaaa

It is easy to show that such a list of strings will contain exactly $\frac{n \cdot (n-1)}{2}$ strings.

You are given n $(n > 2)$ and k $(1 \le k \le \frac{n \cdot (n-1)}{2})$. Print the k-th string from the list.

a a a b b
a a b a b
a a b b a
a b a a b
a b a b a
a b b a a
b a a a b
b a a b a
b a b a a
b b a a a

} → shows lexicographical order

Have to find K_th value.

Suppose we want to find ont if the K-th string will have 'a' in the beginning or not.

$\boxed{a}$ If I fix 'a', then how many strings are possible that will have a in the beginning.

$\boxed{a}$ _ _ _ _ 4Place 2B

⇓

$^4C_2 = 6$

6 string that will have a in the beginning.
If my K<6 then a can be fixed.

→ K=3

a $\boxed{a}$ → check $^3C_2 = 3$ (so I can put)

a a $\boxed{a}$ → check $^2C_2 = 1$ (But K=3) so no a, ∴ Put b

lexicographically smaller than above string ↗

a a b

K will become = 3 − (no. of strings)

→ K=2

Starting from aab, have to find 2nd lexicographical string

a a b $\boxed{a}$ $^1C_1$ = (But K=2) So Put b

aabb

Now K=1

⇒ aabba **Ans**

```
using namespace std;

#define forn(i, n) for (int i = 0; i < int(n); i++)

int main() {
    int t;
    cin >> t;
    forn(tt, t) {
        int n, k;
        cin >> n >> k;
        string s(n, 'a');
        for (int i = n - 2; i >= 0; i--) {
            if (k <= (n - i - 1)) {
                s[i] = 'b';
                s[n - k] = 'b';
                cout << s << endl;
                break;
            }
            k -= (n - i - 1);
        }
    }
}
```

Ques. 5 Leha & Function

https://codeforces.com/problemset/problem/840/A

Leha like all kinds of strange things. Recently he liked the function $F(n, k)$. Consider all possible k-element subsets of the set $[1, 2, ..., n]$. For subset find minimal element in it. $F(n, k)$ — mathematical expectation of the minimal element among all k-element subsets.

But only function does not interest him. He wants to do interesting things with it. Mom brought him two arrays A and B, each consists of m integers. For all i, j such that $1 \le i, j \le m$ the condition $A_i \ge B_j$ holds. Help Leha rearrange the numbers in the array A so that the sum $\sum_{i=1}^{m} F(A'_i, B_i)$ is maximally possible, where A' is already rearranged array.

F(n,K)

Set = [1,2,3 _ _ _ n]

Subset of length = K

How many subset possible = ^nC_K

Dice Throw : 1, 2, 3, 4, 5, 6

What is the expected value if you toss a dice?

$$\Rightarrow \Sigma P[x_i] * x_i$$

In the above case of dice

$$= \frac{1}{6} \Sigma 1+2+3+ \dots +6$$

So, in our original question:

$n = [1, 2, \dots n]$

(length = k) ⇒ minimal value

$$E = \frac{1}{{}^{n}C_{k}} * \text{minimum val in that subset}$$

$$\Rightarrow \frac{1}{{}^{n}C_{k}} \Sigma \text{min val}$$

Use of contribution Technique

$$\Rightarrow \frac{1 * {}^{n-1}C_{k-1} + 2 * {}^{n-2}C_{k-1} + 3 * {}^{n-3}C_{k-1} + \dots}{{}^{n}C_{k}}$$

[Hockey Stick : Method to simplify Binomial Expression]