CSP-J 数学

crazy_cloud

SJTU

2022 年 10 月 5 日

目录

目录

数学内容在 CSP-J 中考察主要以简单的取模运算和组合计数为主。

表: 近五年考察内容

年份	题目名	考查内容
2019（江西）	次大值	取模运算，分类讨论
2019（江西）	非回文串	组合计数，补集转换
2021	分糖果	取模运算，分类讨论

目录

基本概念

约数和因子

如果 $d|a$ 且 $d \geq 0$，则称 d 是 a 的约数。
正整数 a 的平凡约数为 1 和 a 本身，a 的非平凡约数称为 a 的因子。

质数与合数

$a > 1$ 且只能被平凡约数整除的数称为质数，$a > 1$ 且不是质数的数成为合数。

同余

两个整数 a 和 b，若它们除以正整数 n 所得的余数相等，则称 a, b 在模 n 意义下同余，记为 $a \equiv b \pmod{n}$。

取模运算及其性质

带余除法

设 a, b 为两个给定的整数，$a \neq 0$。那么，一定存在唯一的一对整数 q 和 r，满足 $b = qa + r, 0 \leq r < |a|$。
我们称 r 为余数。
定义 $b \bmod a := r$。

例

如果 $b > a > 0$，则 $b \bmod a \leq \lfloor b/2 \rfloor$。

最大公约数和最小公倍数

最大公约数

两个不同时为 0 的 a 与 b 的公共约数中最大的称为 a 与 b 的最大公约数，记作 $\gcd(a, b)$。

最小公倍数

两个不同时为 0 的 a 与 b 的公共倍数中最小的称为 a 与 b 的最小公倍数，记作 $\operatorname{lcm}(a, b)$。

总所周知，$\gcd(a, b) \operatorname{lcm}(a, b) = ab$。

欧几里得算法

原理：$\gcd(a, b) = \gcd(b, a - b)$。

Euclidean algorithm

```
function GCD(a, b)
    if b = 0 then
        return a
    else
        return GCD(b, a mod b)
    end if
end function
```

每一步会使得其中一个参数减半，总的时间复杂度是 $O(\log n)$。

目录

每个合数 a 一定能够写成 $a = px(p \in \mathbb{P}, x > 1)$ 的形式。
对于每一个 n 以内的质数 p，枚举倍数 x，把 px 标记为合数。
筛选的时候可以做一个小的改进：对于质数 p，只筛选倍数 $x \geq p$ 的数，因为如果 $x < p$，则 x 中一定有比 p 小的质因子，px 就会在前面的筛选过程中被筛出。
于是只需要考虑 $\sqrt{n}$ 以内的质数就好了。
时间复杂度 $O(n \ln \ln n)$，不证。

Sieve of Eratosthenes

procedure $\textsc{SieveOfEratosthenes}(n)$
  set all *isPrime* to be true
  $isPrime[1] \leftarrow$ false
  for $p = 2, \ldots, \sqrt{n}$ **do**
    if $isPrime[p]$ =true **then**
      for $x = p, \ldots, \lfloor n/p \rfloor$ **do**
        $isPrime[px] \leftarrow$ false
      end for
    end if
  end for
end procedure

目录

例 (CSP-J2021 - 分糖果)

给定正整数 n, L, R，求 $\max_{L\le k\le R}\{k \bmod n\}$。
$2 \le n \le L \le R \le 10^9$。

如果 $\lfloor L/n \rfloor = \lfloor R/n \rfloor$，则 $[L, R]$ 里面所有数模 n 的结果是单调递增的，于是 $k = R$ 时 $k \bmod n$ 最大。
否则，$(L, R]$ 中至少存在一个 n 的倍数（记为 N），当 $k = N - 1$ 时，$k \bmod n$ 最大，为 $n - 1$。

例 (number)

给定 L, R，计算

$$\max_{L\le i,j\le R} \gcd(i,j),$$

$$\min_{L\le i,j\le R} \gcd(i,j)$$

$$\max_{L\le i,j\le R} \operatorname{lcm}(i,j),$$

$$\min_{L\le i,j\le R} \operatorname{lcm}(i,j).$$

保证 $L, R < 2^{31}$。

例 (number)

给定 L, R，计算

$$\max_{L\le i,j\le R} \gcd(i,j),$$

$$\min_{L\le i,j\le R} \gcd(i,j)$$

$$\max_{L\le i,j\le R} \operatorname{lcm}(i,j),$$

$$\min_{L\le i,j\le R} \operatorname{lcm}(i,j).$$

保证 $L, R < 2^{31}$。

当 $L < R$ 时，存在互质数对（相邻）。同时注意 i 与 j 可以取同一个数（尤其是 L 或 R）。

例 (CSP-J2019 JX - 次大值)

给定 n 个正整数 $a_1, a_2, \ldots, a_n$。
请问所有 $a_i \bmod a_j (1 \leq i, j \leq n$ 且 $i \neq j)$ 去重后次大值是多少。
$3 \leq n \leq 2 \times 10^5, 1 \leq a_i \leq 10^9$。

首先我们去掉重复的数。
从小到大排序之后，最大的一定是 $a_{n-1} \bmod a_n$。

例 (CSP-J2019 JX - 次大值)

给定 n 个正整数 $a_1, a_2, \ldots, a_n$。
请问所有 $a_i \bmod a_j (1 \le i, j \le n$ 且 $i \ne j)$ 去重后次大值是多少。
$3 \le n \le 2 \times 10^5, 1 \le a_i \le 10^9$。

分类讨论。
一个数模比它大的数，那么 $a_{n-2} \bmod a_n$ 肯定是最佳候选。

次大值

例 (CSP-J2019 JX - 次大值)

给定 n 个正整数 $a_1, a_2, \ldots, a_n$。
请问所有 $a_i \bmod a_j (1 \le i, j \le n$ 且$i \ne j)$ 去重后次大值是多少。
$3 \le n \le 2 \times 10^5, 1 \le a_i \le 10^9$。

一个数模比它小的数，那我们肯定不能模小于等于 a_{n-2} 的，那只 $a_n \bmod a_{n-1}$。
两者取最大即可。

互质

例 (Luogu P1592 - 互质)

求与 n 互质的第 k 个正整数。
$1 \leq n \leq 10^6, 1 \leq k \leq 10^8$。

$\forall m \in \mathbb{N}, \gcd(n, i) = \gcd(n, nm + i)$。
只需要预处理 n 以内和 n 互质的数，就能直接推算出第 k 个。

例 (CF 1656C - Make Equal With Mod)

给定 n 个非负整数 $a_1, a_2, \ldots, a_n$。
每次操作，你可以选定一个整数 $x \geq 2$，将所有 a_i 替换成 $a_i \bmod x$。
判断是否有可能通过若干次操作将所有 a_i 变得相等。
$1 \leq n \leq 10^5, 0 \leq a_i \leq 10^9$。

一个自然而然的想法，从大到小每次取 $x = a_i$，把所有数都变成 0。
要求 $x > 1$，当存在 $a_i = 1$ 的时候怎么办？
这种情况下，我们只能考虑把所有数都变成 1。
又一个自然而然的想法，从大到小每次取 $x = a_i - 1$。
唯一不能 work 的情况，存在 $a_i = a_{i-1} + 1$？
可以发现，这个时候我们无论怎么模，这两个数都不可能相等。所以无解。

目录

加法原理和乘法原理

加法原理

完成一个事情可以有 n 类方法，$a_i(1 \leq i \leq n)$ 表示第 i 类方法的数目。
那么完成这件事情的不同的方法数是 $S = a_1 + a_2 + \cdots + a_n$。

乘法原理

完成一件事情需要 n 个步骤，$a_i(1 \leq i \leq n)$ 表示第 i 个步骤的不同方法数目。
那么完成这件事情共有 $S = a_1 \times a_2 \times \cdots \times a_n$ 种不同的方法。

容斥原理和补集转换

补集转换是容斥原理的简单应用，但是比起容斥的原始形式更加常用。
补集转换的意思就是，我们现在需要求出满足条件的方案数，但是这个东西不好求。
那么我们就求出不满足条件的方案数，然后用总情况数减去这个，就得到了答案。

例 (实验舱 780 - 越狱)

监狱有连续编号为 1 到 n 的 n 个房间，每个房间关押一个犯人。有 m 种宗教，每个犯人可能信仰其中一种。如果相邻房间的犯人信仰的宗教相同，就可能发生越狱。求有多少种状态可能发生越狱。

补集转换，总的方案数是 m^n，不发生越狱的方案数是 $m \times (m-1)^{n-1}$，两者相减即可。

抽屉原理

抽屉原理

将 n 个物体，划分为 k 组，那么至少存在一个分组，含有大于或等于 $\left\lceil \frac{n}{k} \right\rceil$ 个物品。

抽屉原理也被称作鸽巢原理。

例 (POJ 2356 - Find a multiple)

给你 n 个数，要你从 n 个数选出若干个数，要求这若干个数的和是 n 的倍数，输出选择数的个数，以及相应的数。

求出这 n 个数的前缀和 $sum_1, sum_2, \cdots, sum_n$。
如果这里面有 n 的倍数，那我们直接输出对应前缀就好了。
否则，$sum_1, sum_2, \cdots, sum_n$ 这 n 个数模 n 的结果在 $[1, n-1]$ 内，抽屉原理，一定存在两个不同的下标 $i < j$，满足 $sum_i = sum_j$。
输出区间 $(i, j]$ 即可。

目录

排列与组合

排列数

从 n 个不同元素中，任取 $m(m \leq n)$ 个元素按照一定顺序排成一排的方案数，叫做排列数，记作 A_n^m。

$$\mathrm{A}_n^m = n(n-1)(n-2)\cdots(n-m+1) = \frac{n!}{(n-m)!}$$

组合数

从 n 个不同元素中，任取 $m(m \leq n)$ 个元素组成一个集合的方案数，叫做组合数，记作 C_n^m 或者 $\binom{n}{m}$，后面统一采用后者。

$$\binom{n}{m} = \frac{\mathrm{A}_n^m}{m!} = \frac{n!}{m!(n-m)!}$$

特殊地，当 $n < m$ 时，我们规定 $\binom{n}{m} = 0$。

多重组合数

多重组合数

多重集是指包含重复元素的广义集合。设 $S=\{n_1\cdot a_1, n_2\cdot a_2, \cdots, n_k\cdot a_k\}$ 表示由 n_1 个 a_1， n_2 个 a_2， ……， n_k 个 a_k 组成的多重集。
则 S 的全排列个数是 $\frac{n!}{\prod_{i=1}^{k} n_i!}$。
令 $n=\sum_{i=1}^{k} n_i$，则记多重组合数

$$\binom{n}{n_1, n_2, \cdots, n_k} = \frac{n!}{\prod_{i=1}^{k} n_i!}$$

杨辉三角

							1							
						1		1						
					1		2		1					
				1		3		3		1				
			1		4		6		4		1			
		1		5		10		10		5		1		
	1		6		15		20		15		6		1	
1		7		21		35		35		21		7		1

组合数的递推公式：$\binom{n}{m} = \binom{n-1}{m-1} + \binom{n-1}{m}$。
组合意义：考虑第 n 个元素是否被选取到集合中。

二项式定理

对于任意自然数 n，有

$$(a+b)^n = \sum_{i=0}^{n} \binom{n}{i} a^{n-i} b^i$$

证明：考虑组合数的递推公式。

例

请计算

$$\sum_{i=0}^{n} \binom{n}{i}$$

并说出它的组合意义。

例

请计算

$$\sum_{i=0}^{n} \binom{n}{i} (-1)^i$$

组合恒等式

例

$$\binom{n}{r}\binom{r}{k} = \binom{n}{k}\binom{n-k}{r-k}$$

组合恒等式

例

$$\sum_{i=0}^{n} i\binom{n}{i} = n2^{n-1}$$

组合恒等式

例

$$\sum_{i=0}^{r} \binom{n}{i} \binom{m}{r-i} = \binom{n+m}{r}$$

组合恒等式

例

$$\sum_{i=0}^{n} \binom{n}{i}^2 = \binom{2n}{n}$$

目录

例 (CSP-J2019 JX - 非回文串)

有 n 个字符，都是小写拉丁字母，分别为 $c_1, c_2, \ldots, c_n$。
将这些字符重新排列，组成一个字符串 S。求有多少种方案能使得 S 不是一个回文串。
只要存在字符编号不同就是不同的排列。
答案对 10^9+7 取模。
$3 \le n \le 2000$。

正难则反，我们考虑计算回文串数目，用 $n!$ 减去这个数目得到答案。
如何计算回文串的数目呢？

非回文串

例 (CSP-J2019 JX - 非回文串)

有 n 个字符，都是小写拉丁字母，分别为 $c_1, c_2, \dots, c_n$。
将这些字符重新排列，组成一个字符串 S。求有多少种方案能使得 S 不是一个回文串。
只要存在字符编号不同就是不同的排列。
答案对 10^9+7 取模。
$3 \le n \le 2000$。

如果有超过一个字母出现了奇数次，那肯定没有办法组成回文串。
下面我们只讨论所有字母均偶出现的情况，存在一个奇出现的情况类似。
设 $a_i (1 \le i \le 26)$ 表示每种字母的个数。
我们先把同种字母看成等价，统计不同的方案。由于左右对称，其实就是一个多重组合数

$$\binom{n/2}{a_1/2, a_2/2, \dots, a_{26}/2}$$

非回文串

例 (CSP-J2019 JX - 非回文串)

有 n 个字符，都是小写拉丁字母，分别为 $c_1, c_2, \ldots, c_n$。
将这些字符重新排列，组成一个字符串 S。求有多少种方案能使得 S 不是一个回文串。
只要存在字符编号不同就是不同的排列。
答案对 $10^9 + 7$ 取模。
$3 \le n \le 2000$。

随后我们将同种字母随意排列，也就是给方案乘上 $\prod_{i=1}^{26} a_i!$。
所以回文串数目为

$$(n/2)! \prod_{i=1}^{26} \frac{a_i!}{(a_i/2)!}$$

这里不需要做除法，直接抵消相同部分后累乘剩下的就好。