

数学

Math

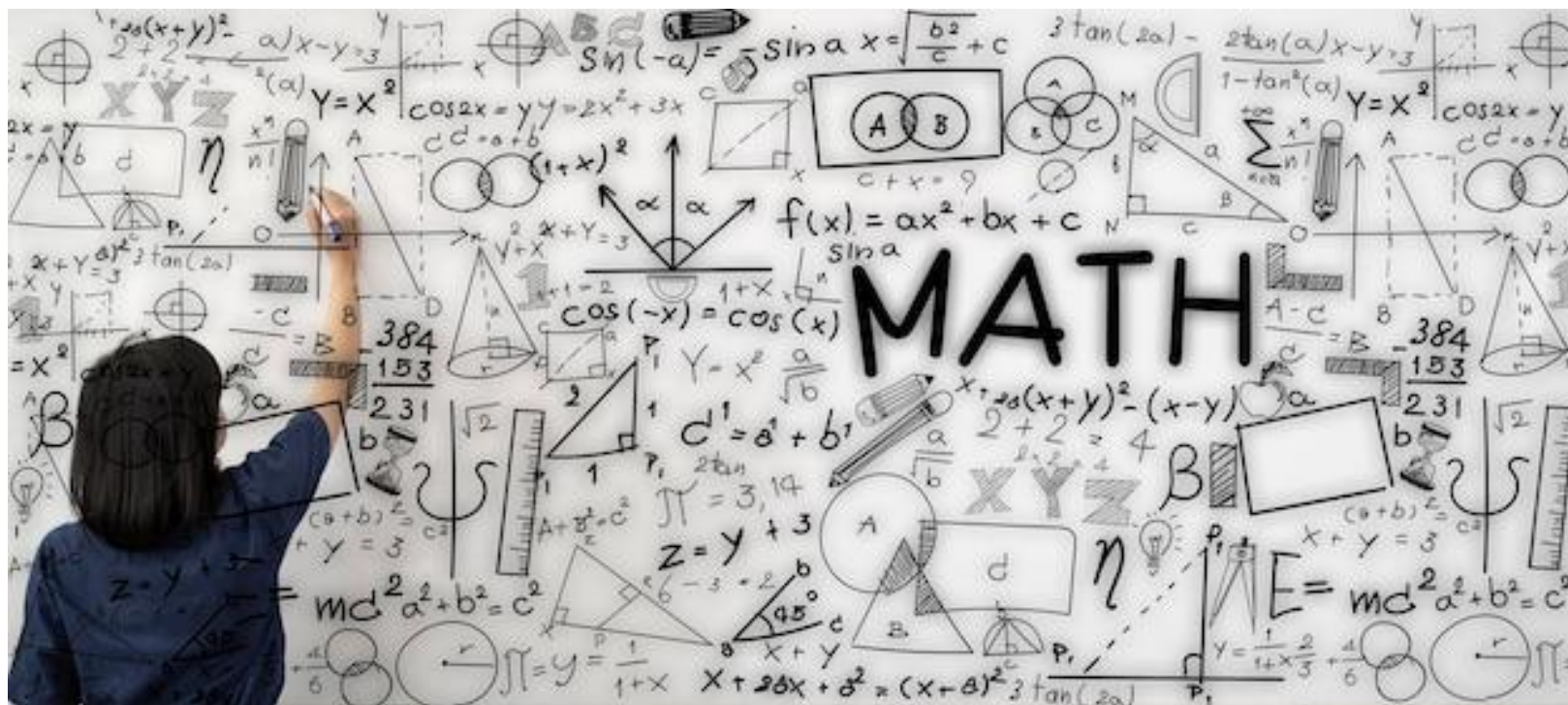
数学是什么？

- **Mathematics** is a field of study that discovers and organizes methods, theories and theorems that are developed and proved for the needs of empirical sciences and mathematics itself.

——Wikipedia

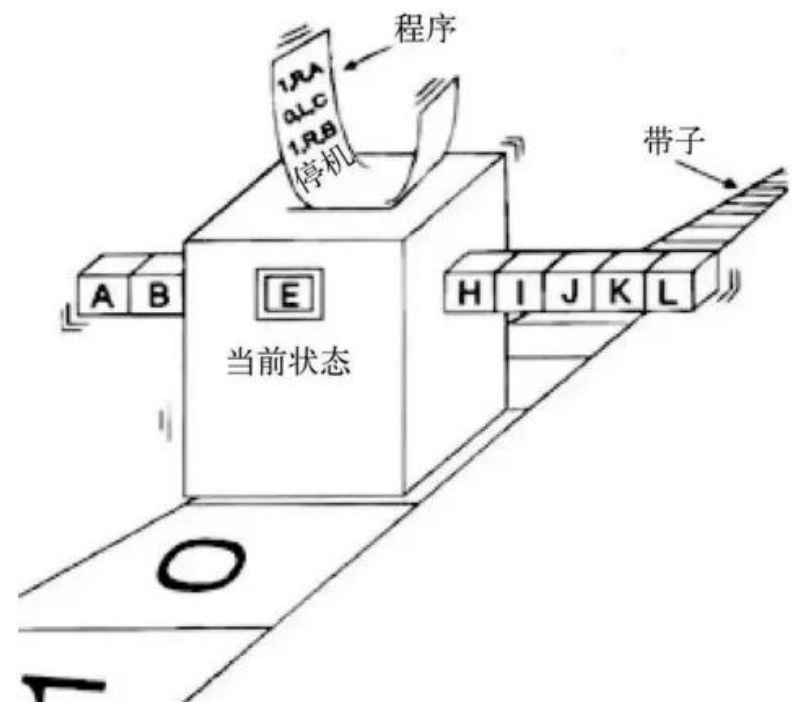
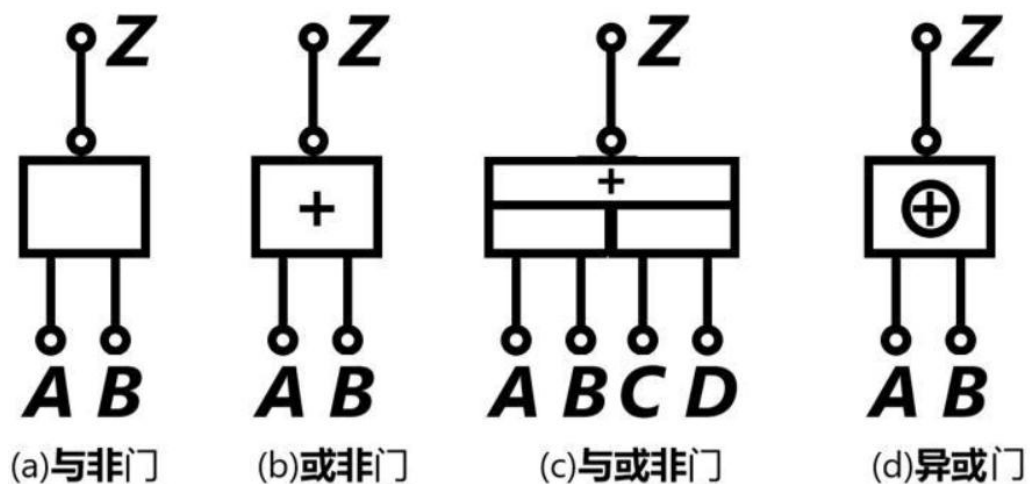
数学是方法和定理的发现和组织， 用来证明科学和数学它本身。

人类来做数学



富有创造力， 但速度慢

计算机来做数学



没有创造力，但速度极快
 $1\text{秒} \approx 10^8\text{次计算}$

OI中常见的数学

- 快速幂
- 素数（质因数分解、筛素数、费马小定理、莫比乌斯反演……）
- 最大公约数、最小公倍数（GCD、LCM）
- 扩展欧几里得（EXGCD）
- 逆元、模数
- 中国剩余定理（CRT）
- 线性代数（矩阵快速幂、高斯消元）
- 递归数、卡特兰数
- 多项式（拉格朗日插值、快速傅里叶变换FFT）

OI中常见的数学

- 快速幂
- 素数（质因数分解、筛素数、费马小定理、莫比乌斯反演……）
- 最大公约数、最小公倍数（GCD、LCM）
- 扩展欧几里得（EXGCD）
- 逆元、模数
- 中国剩余定理（CRT）
- 线性代数（矩阵快速幂、高斯消元）
- 递归数、卡特兰数
- 多项式（拉格朗日插值、快速傅里叶变换FFT）

模数（同余）

$$a \bmod b = a - \left\lfloor \frac{a}{b} \right\rfloor \cdot b$$

1. $(a + b) \bmod p = (a \bmod p + b \bmod p) \bmod p$

2. $(a - b) \bmod p = (a \bmod p - b \bmod p) \bmod p$

3. $(a \times b) \bmod p = (a \bmod p) \times (b \bmod p) \bmod p$

4. 若 $a \equiv b \pmod{p}$, 那么 $a^n \equiv b^n \pmod{p}$

5. 若 $a \bmod p_1 = x$, $a \bmod p_2 = x$, 且 p_1, p_2 互质, $a \bmod (p_1 \times p_2) = x$

6. 若 $a \equiv b \pmod{p}$, k 和 c 为整数, 而且 $k > 0$, 那么 $a^k c \equiv b^k c \pmod{p}$

7. 若 $ac \equiv bc \pmod{p}$, 那么 $a \equiv b \pmod{\frac{p}{\gcd(p, c)}}$ 就可以推出 $\gcd(p, c) = 1$, 则有 $a \equiv b \pmod{p}$

快速幂

$$a^b \bmod M$$

$$a^b$$

$$= a^{b/2} \cdot a^{b/2}$$

$$5^{11}$$

$$= 5^1 \cdot 5^2 \cdot 5^8$$

```
LL fastPow(LL a, LL n, LL m)
{
    LL ans = 1;
    while(n)
    {
        if(n & 1) ans *= a, ans %= m;
        a *= a, a %= m, n >>= 1;
    }
    return ans % m;
}
```

P1226 【模板】快速幂

GCD和LCM

- 最大公约数（GCD）：多个数共有的约数中最大的那个
- 最小公倍数（LCM）：多个数共有的倍数中最小的那个
- 求GCD：辗转相除法（欧几里得算法） $\gcd(a, b) = \gcd(b, a \bmod b)$

```
int gcd(int a, int b)
{
    if(!b) return a;
    else return gcd(b, a % b);
}
```

$$\text{lcm}(a, b) = \frac{ab}{\gcd(a, b)}$$

素数-质因数分解

- 任何一个正整数都可以表示成多个素数的幂的和

$$n = \sum_{i=1}^k p_i^{m_i}$$

- 考虑 $O(\sqrt{n})$ 的做法

```
void Factorize(LL n, vector<LL> &factors, map<LL,LL> &primeCnt)
{
    for(LL i = 2; i * i <= n; i++)
    {
        if(n % i == 0)
        {
            factors.push_back((LL)i);
            while (n % i == 0)
            {
                primeCnt[(LL)i]++;
                n /= i;
            }
        }
    }
    if(n > 1)
    {
        factors.push_back((LL)n);
        primeCnt[(LL)n]++;
    }
}
```

B3871 因数分解

素数-筛素数

- 有人问:

老师, 你的 $O(\sqrt{n})$ 判断单个素数确实很快, 但我想快速找出1-n里的所有素数, 有什么办法吗?

- 有的兄弟, 有的

- 欧拉线性筛素数 $O(n)$

```
bool isPrime[MAXN];
int Prime[MAXN];

void EulerPrime(int n)
{
    int cnt = 0;
    memset(isPrime, true, sizeof(isPrime));
    isPrime[1] = 0;
    for(int x = 2; x <= n; x++)
    {
        if(isPrime[x]) Prime[++cnt] = x;
        for(int y = 1; y <= cnt && x*Prime[y] <= n; y++)
        {
            isPrime[x*Prime[y]] = false;
            if(x % Prime[y] == 0) break;
        }
    }
}
```

素数-费马小定理

- 我们有整数 a 和**素数** p , 则

$$a^p \equiv a \pmod{p}$$

- 如果 a 不是 p 的倍数, 我们又有

$$a^{p-1} \equiv 1 \pmod{p}$$

- 如果 a 是 p 的倍数呢?

$$a^{p-1} \equiv 0 \pmod{p}$$

逆元

- 逆元素
- 对于一个数 a ，它的加法逆元就是 $-a$
- 乘法逆元? $\frac{1}{a}$
- 模意义下的乘法逆元? ? $\frac{1}{a} \equiv ? \pmod{p}$

逆元

$$a^{p-1} \equiv 1 \pmod{p}$$

- 考虑继续变形

$$a^{p-2} \equiv \frac{1}{a} \pmod{p}$$

- 也就是说，对于单个 a ，在模素数 p 意义下的乘法逆元就是

`fastPow(a, p-2, p);`

训练赛

10:30-12:00