# 数学 06

- 前置知识
- 欧拉函数
- 欧拉定理
- 题目选讲

# 前置知识

在介绍欧拉函数和欧拉定理之前，我们需要先介绍一些数学名词

- 积性函数指的是满足 $(a,\ b) = 1,\ f(ab) = f(a) \cdot f(b)$ 性质的函数
- 模 $p$ 的完全剩余系表示 $\{1,\ 2,\ ..,\ p-1\}$
- 模 $p$ 的简化剩余系表示不超过 $p$ 且与 $p$ 互质的数

# 欧拉函数

给定正整数 $m$，我们有时候会关心 $1 \sim m$ 中与 $m$ 互质的正整数，因此有了欧拉函数的定义：

$m$ 的欧拉函数表示不超过 $m$ 且与 $m$ 互质的正整数个数，记作 $\phi(m)$

例如，$\phi(1) = 1$，$\phi(3) = 2$，$\phi(10) = 4$

# 欧拉函数长什么样

$$\phi(m) = m\prod_{i=1}^{n}(1 - \frac{1}{p_i})$$

其中 $p_i$ 是 $m$ 的质因子，$m$ 一共有 $n$ 个不同的质因子

我们可以验证一下这个表达式，例如 $m = 12$

根据定义，那么不超过 12 且与 12 互质的数有 1，5，7，11，共计 4 个，因此 $\phi(12) = 4$

根据表达式，$m = 12 = 2^2 \times 3$，有两个质因子 2，3，那么 $\phi(12) = 12 \times (1 - \frac{1}{2}) \times (1 - \frac{1}{3}) = 4$

可以动手算一下一些欧拉函数，例如 $\phi(15), \phi(27)$，$\phi(50)$

表达式的证明在 notes.md 中，有兴趣的同学可以阅读

# 写程序计算欧拉函数

下面把数学推导化成程序代码

# 求单个数的欧拉函数

给定 $m$，如何计算 $\phi(m)$?

根据

$$\phi(m) = m\prod_{i=1}^{n}(1 - \frac{1}{p_i})$$

我们考虑先筛出质因子，在利用公式求解，当然这个过程可以同时进行，具体代码如下所示

```cpp
int phi(int m)
{
    int ans = m;
    for(int i = 2; i * i <= m; ++i) {
        if(m % i == 0) {
            ans -= ans / i;
            while(m % i == 0) m /= i;
        }
    }
    if(m > 1) ans -= ans / m;
    return ans;
}
```

# 线性筛一个范围内的欧拉函数

我们知道，在欧拉筛中，合数 $n$ 被其最小的质因子 $p$ 筛去

设 $n = n' \cdot p$，用 $\mid$ 表示整除

若 $p \mid n'$，则 $n'$ 含有 $n$ 的所有质因子

所以

$$\phi(n) = p \cdot n' \cdot \prod_{i=1}^{n}(1 - \frac{1}{p_i}) = p \cdot \phi(n')$$

若 $p \nmid n'$，则 $(p,\ n') = 1$，由积性函数性质得到

$$\phi(n) = \phi(p) \cdot \phi(n') = (p-1) \cdot \phi(n')$$

因此我们得到一个递推公式

$$\phi(n) = \begin{cases} p \cdot \phi(\frac{n}{p}), & p \mid \frac{n}{p} \\ (p-1) \cdot \phi(\frac{n}{p}), & p \nmid \frac{n}{p} \end{cases}$$

我们只需要对欧拉筛稍作变动，便可以得到期望的程序

```cpp
int n, prime[N], phi[N], vis[N];
int Euler_sieve()
{
    int cnt = 0;
    for(int i = 2; i <= n; ++i) vis[i] = 1;
    for(int i = 2; i <= n; ++i){
        if(vis[i]) prime[++cnt] = i, phi[i] = i - 1; //素数的欧拉函数值为 i - 1
        for(int j = 1; j <= cnt && prime[j] * i <= n; ++j){
            vis[prime[j] * i] = 0;
            phi[prime[j] * i] = (prime[j] - 1) * phi[i];
            if(i % prime[j] == 0) {
                phi[prime[j] * i] = prime[j] * phi[i];
                break;
            }
        }
    }
    return cnt;
}
```

# 欧拉定理

聊完欧拉函数，我们来讨论欧拉定理

# 什么是欧拉定理

用 $(a,\ b)=1$ 表示 $a,\ b$ 互质

欧拉定理说，若 $(a,\ m)=1$，则 $a^{\phi(m)}\equiv 1\ (mod\ m)$

举个例子，$a=5,\ m=12$，根据前面的计算我们知道 $\phi(m)=4$，因此 $5^4\equiv 1\ (mod\ 12)$，这与直接计算的结果是一样的

欧拉定理的证明在 [notes.md]()，有兴趣的同学可以阅读

# 费马小定理

如果你对快速幂还有印象，那你一定记得费马小定理这个名词，费马小定理说的是这样一件事：

若 $p$ 是质数，给定正整数 $a$，则 $a^p \equiv a \ (mod \ p)$，也即是 $a^{p-1} \equiv 1 \ (mod \ p)$

事实上，由于 $p$ 是质数，因此 $\phi(p) = p - 1$，因此费马小定理是欧拉定理的一个特例

# 拓展欧拉定理

欧拉定理是局限于底数与模数互质，为了打破这个限制，我们引入拓展欧拉定理

对于任意的正整数 $a$，$b$，$m$，有

$$a^b \equiv a^{b \; mod \; \phi(m)+\phi(m)} \; (mod \; m)$$

更具体的有

$$a^b \equiv \begin{cases} a^{b \bmod \phi(p)}, & (a,\ p) = 1 \\ a^b, & (a,\ p) \neq 1,\ b < \phi(p) \\ a^{b \bmod \phi(p) + \phi(p)}, & (a,\ p) \neq 1,\ b \geq \phi(p) \end{cases} \pmod p$$

证明极其复杂，不在这里列出

# 欧拉定理的用途

讲了这么多欧拉定理，大家一定很关心它的用途

考虑这样一个问题，计算

$$3^{134725498254347528934571264577789237478619831 2} \; mod \; 11$$

你会发现指数太大了，如果要存储，就要用到高精度，然后进行高精度下的快速幂，这是可以的，但是时间复杂度很高

因此我们可以用欧拉定理对指数降幂，把指数降下来再使用快速幂，这就是 **欧拉降幂**

# 1471. 欧拉函数

给定 $n$，计算 $\phi(2) \oplus \phi(3) \oplus .. \phi(n)$，其中 $\oplus$ 表示异或

规定 $1 \leq n \leq 1,000,000$

# 题解

这个数据范围可以使用线性筛，然后异或得到答案

# 2609. 欧拉降幂

给定 $a$，$m$，$b$，计算 $a^b \ mod \ m$

输入保证

$1 \leq a \leq 1,000,000,000$

$1 \leq m \leq 100,000,000$

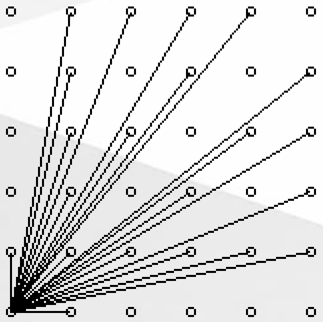
$1 \leq b \leq 10^{100,000}$

# 题解

使用扩展欧拉定理降幂，把 $b$ 当作字符串读入，边读入边取模，最后求一个快速幂

# 2610 仪仗队

给定 $N \times N$ 的仪仗队，$C$ 站在左下角，计算他所看到的人数



$1 \leq N \leq 40,000$

# 题解

设 $dp_i$ 表示从 $i \times i$ 个人中看到的人数

考虑再往外延展一层，由于对称性，只需要考虑横着增加的 $i + 1$ 个人

以左下角为原点建立直角坐标系，设某个同学坐标为 $(x, y)$

- 若 $x \mid y$，则他不会被看到，因为在他之前有一个人已经被看到了，那个人的坐标为 $\left(\frac{x}{(x, y)}, \frac{y}{(x, y)}\right)$
- 若 $x \nmid y$，那么他会被看到，因为前面没有人挡住

对于新增的一层，共有 $0$，$1$，..，$i$ 等 $i+1$ 个横坐标，纵坐标为 $i$，那么一共新增了 $\phi(i)$ 个人

竖直方向同理，因此一共新增 $2\phi(i)$ 个人

所以 $dp_i = dp_{i-1} + 2\phi(i-1)$