欧拉函数 - CSDN博客

欧拉函数的定义:E(k)=([1,n-1]中与n互质的整数个数).

因为任意正整数都可以唯一表示成如下形式:

k=p1^a1*p2^a2*……*pi^ai;(即分解质因数形式)

可以推出:E(k)=(p1-1)(p2-1)……(pi-1)*(p1^(a1-1))(p2^(a2-1))……(pi^(ai-1))

=k*(p1-1)(p2-1)……(pi-1)/(p1*p2*……pi);

=k*(1-1/p1)*(1-1/p2)....(1-1/pk)

ps：在程序中利用欧拉函数如下性质，可以快速求出欧拉函数的值(a为N的质因素)

若(N%a==0 && (N/a)%a==0) 则有:E(N)=E(N/a)*a;

若(N%a==0 && (N/a)%a!=0) 则有:E(N)=E(N/a)*(a-1);

http://hi.baidu.com/ldante/blog/item/996b0ea131a7a58f46106443.html

第一次写欧拉函数的题，琢磨的半天，最后还是只能按照最开始的想法写......

欧拉函数PHI(n)表示的是比n小，并且与n互质的正整数的个数(包括1)。比如：

PHI(1) = 1; PHI(2) = 1; PHI(3) = 2; PHI(4) = 2; ... PHI(9) = 6; ...

要计算一个正整数n的欧拉函数的方法如下：

1. 将n表示成素数的乘积: n = p1 ^ k1 * p2 ^ k2 * ... * pn ^ kn（这里p1, p2, ..., pn是素数）

2. PHI(n) = (p1 ^ k1 - p1 ^ (k1 - 1)) * (p2 ^ k2 - p2 ^ (k2 - 1)) * ... *(pn ^ kn - pn ^ (kn - 1))

= Mult { pi ^ ki - pi ^ (ki -1) }

证明过程如下：

1. 容易想到：当n为素数时，PHI(n) = n - 1。因为每个比n小的正整数都和n互素。当n为素数p的k次方时，PHI(n) = p ^ k - p ^ (k - 1)。因为在1到n之间的正整数只有p的倍数和n不互素，这样的数有(p ^ k / p)个。

2. 如果m和n互素，即GCD(m, n) = 1，那么PHI(m * n) = PHI(m) * PHI(n)。用中国剩余定理可以证明，证明的思路是建立这样一种一一对应的关系(a, b) <-> x，其中正整数a小于m并且gcd(a, m) = 1，正整数b小于n并且gcd(b, n) = 1，正整数x小于m*n并且gcd(m*n, x) = 1。证明过程如下：

1)根据中国剩余定理，如果m和n互素，那么关于未知量x的方程组x % m = a, x % n = b(0 <= a < m, 0 <= b < n)，当0 <= x < m * n时存在并且仅存在一个解。容易证明，如果两个这样的方程组有相同的m, n但是a, b不同，那么他们的解x一定不同。

2)首先用反正法证明：gcd(m, a) = 1且gcd(n, b) = 1是gcd(m*n, x) = 1的必要条件：假设gcd(a, m) = k > 1，由此可得：a = a' * k; m = m' * k => x = k' * m + a = k' * k * m' + k * a' = k * (k' * m' + a'); 所以gcd(x, m) = k > 1。同理可证，如果gcd(b, n) > 1， 那么gcd(x, n) > 1。所以x和m * n互素的必要条件是a和m互诉且b和n互素。

3)接下来我们证明充分性：由x % m = a 可以得到x = k * m + a；由欧几里德算法求最大公约数的过程（就不证明了，呵呵，还得想）可以知道gcd(x, m) = gcd(m, a) = 1；同理可得，如果gcd(n, b) = 1那么gcd(x, n) = 1。接下来很容易得到：gcd(m*n, x) = 1。从而证明了充分性。

4)上面三步的结论表明，数对(a, b)是可以和x建立起一一对应的关系的，所以有多少个不同的(a, b)，就有多少个不同的x。

3.将n分解成素数乘积后，显然对于任意的i, j(i != j)都满足 pi ^ ki和pj ^ kj是互素的，于是可以的到上面的公式。

跟据上面的公式，可以得到关于欧拉函数的递推关系：

假设素数p能整除n，那么

如果p还能整除n / p, PHI(n) = PHI(n / p) * p;

如果p不能整除n / p, PHI(n) = PHI(n / p) * (p - 1);

下面是两种求欧拉函数的不同编程方法：

/*==*\

|递推求欧拉函数phi(i)

*==

```
==============================*/
for (i = 1; i <= maxn; i++) phi[i] = i;
for (i = 2; i <= maxn; i += 2) phi[i] /= 2;
for (i = 3; i <= maxn; i += 2) if (phi[i] == i) {
for (j = i; j <= maxn; j += i)
phi[j] = phi[j] / i * (i - 1);
/*============================================================*\
|单独求欧拉函数phi(x)
\*============================================================*/
unsigned euler(unsigned x)
{// 就是公式
unsigned i, res =x;
for (i = 2; i < (int)sqrt(x * 1.0) + 1; i++)
if(x%i==0) {
res = res / i * (i - 1);
while (x % i == 0) x /= i; // 保
```

```
证i一
定是素
数
}
if (x > 1
) res = r
es / x *
(x - 1);
return r
es;
}
```