欧几里得算法与拓展欧几里得算法

Ⅰ.欧几里得算法（GCD）

一. 用途：在 lg(n)的时间复杂度级别，求取两个整数的的最大公因数

二. 相关结论：

1. GCD(a, b) = GCD(b, a % b)

证明：令 a = q * b + r

则要证 GCD(a, b) = GCD(b, b % a)

即证 GCD(a, b) = GCD(b, r)

不妨设 GCD(a, b) = d

即证 d 为 b、r 的最大公因数

a) 证明 d 为 d、r 的公因数：

设 a / d = x

b / d = y

因为 d 为 a、b 的最大公因数，所以 a，b 互质

所以 b = d * y

r = d * (x−n*y)

因为 x,y,n ∈ Z，所以 d 为 b、r 的因数

b) 证明 d 是 b、r 的最大公因数

设 b / d = y = α

r / d = x − n*y = β

所以 α / β = x / y − n

因为 x，y 互质，所以 x / y 不为整数

所以 α 与 β 互质

即 b、r 被 d 除后无公因数

综上，得证

2. GCD(a, 0) = a

证明：这个很显然

三. 代码参考：http://paste.ubuntu.com/17896452/

Ⅱ.拓展欧几里得算法（exGCD）

一. 用途：在 lg(n)的时间复杂度级别，求解 a*x + b*y = GCD(a, b)的一组整数解

二. 算法思路：

对方程的系数进行辗转相除，最后在倒推出 x、y

参见 wiki[1]：https://zh.wikipedia.org/wiki

补充：

对于每一次辗转相除：

由 a*x + b*y = GCD(a, b)

得 (a%b)*x + b*(y+(a/b)*x) = GCD(a, b)

这就是为什么倒推回去的时候 y −= (a/b)*x

三. 代码参考：http://paste.ubuntu.com/17896713/

[1] 讲的非常详细，不是很清楚的同学建议跟着一起手推一推