逆元

1.4 逆元

若 $a*x\equiv1(\mathrm{mod}\ b)$，$a$，$b$ 互质，则称 x 为 a 的逆元，记为 a^{-1}。

根据逆元的定义，可转化为 $a*x+b*y=1$，用拓展欧几里得法求解。逆元可以用来在计算 $(t/a)\ \mathrm{mod}\ b$ 时，转化为 $t*a^{-1}\ \mathrm{mod}\ b$。

利用快速幂及扩展欧几里德算法求逆元的代码如下：

```
void exgcd(int a,int b,int c,int &x,int &y){
        if(a==0){
                x=0;y=c/b;
                return;
        }
```



```
        else{
                int tx,ty;
                exgcd(b%a,a,tx,ty);
                x=ty-(b/a)*tx;
                y=tx;
                return;
        }
}
```

求逆元还有一个线性算法，具体过程如下。

首先，$1^{-1}\equiv1(\mathrm{mod}\ p)$

然后，我们设 $p=k*i+r$，$r<i$，$1<i<p$，再将这个式子放到 mod p 意义下就会得到：

$k*i+r\equiv0(\mathrm{mod}\ p)$

再两边同时乘上 i^{-1}，r^{-1} 就会得到：

$$k*r^{-1}+i^{-1}\equiv0 \quad (\mathrm{mod}\ p)$$

$$i^{-1}\equiv-k*r^{-1} \quad (\mathrm{mod}\ p)$$

$$i^{-1}\equiv-\left[\frac{p}{i}\right]*(p\ \mathrm{mod}\ i)^{-1} \quad (\mathrm{mod}\ p)$$

于是，就可以从前面推出当前的逆元了。代码也就一行：

```
A[i]=-(p/i)*A[p%i];
```

实际上，这也提供了一种 $\Theta(\log_2 p)$ 的时间内求出单个数逆元的方法，只要直接按照那个公式递归就可以了。可以证明：$p\ \mathrm{mod}\ i<i/2$，每次递归问题规模减半，最终只会有 $\Theta(\log_2 p)$ 次递归。