

[报告]poj 3696 The Luckiest Number

[Source] <http://poj.org/problem?id=3696>

[Description]

给一 2,000,000,000 以内的整数 L，问至少多少个 8 连在一起组成的自然数是 L 的倍数。如果不可能有这样的数，输出 0。

[Solution]

注意到凡是那种 1111111..... 2222222..... 33333.....

之类的序列都可用这个式子来表示：

$$k \cdot (10^n - 1) / 9$$

进而简化：

这个题会变成：

$$8 \cdot (10^n - 1) / 9 = k \cdot m \quad (k \text{ 是一个整数})$$

如果  $\gcd(8, m) = t$  的话

那么原始的式子可以改为：

$$8/t \cdot (10^n - 1) = 9m/t \cdot k$$

由于  $8/t$  和  $9m/t$  完全没有公约数

所以如果  $8/t \cdot (10^n - 1)$  要整除  $9m/t$  的话，必须是

$10^n - 1$  是  $9m/t$  的整数倍

转化一下就变成了：

$10^n \equiv 1 \pmod{m}$  的形式

求出  $m$  的欧拉函数值，测试  $\phi(m)$  的每一个约数

PS：上一篇的二分幂取模在这里不能用，因为这个题的数据范围太大了

[Code]

```
#include<cstdio>
#include<cstdlib>
#include<cstring>
```

```

#include<cmath>

#include<algorithm>

#define MAXN 1000000

#define PRIME 80000

typedef unsigned long long INT;

INT isprime[MAXN+10];

INT prime[PRIME+10];

INT top;

void init(void)
{
    INT i,j;

    for(i=2;i<=MAXN;i++)

    {

        if(!isprime[i])    prime[top++]=i;

        for(j=0;j<top&& i*prime[j]<=MAXN;j++)

        {

            isprime[i*prime[j]]=true;

            if(i%prime[j]==0)    break;

        }

    }

    top--;

    isprime[1]=1;
}

INT gcd(INT a,INT b)
{

    if(b==0)

        return a;

    else return gcd(b,a%b);
}

```

```
}
```

```
INT multimod(INT a,INT b,INT n)
```

```
{
```

```
    a=a%n;
```

```
    INT res=0;
```

```
    while(b)
```

```
    {
```

```
        if(b&1)
```

```
        {
```

```
            res=res+a;
```

```
            if(res>=n)
```

```
                res=res-n;
```

```
        }
```

```
        a=a<<1;
```

```
        if(a>=n)
```

```
            a=a-n;
```

```
        b=b>>1;
```

```
    }
```

```
    return res;
```

```
}
```

```
INT quickmod(INT a,INT n,INT p)
```

```
{
```

```
    a=a%p;
```

```
    INT ans=1;
```

```
    while(n>=1)
```

```
    {
```

```
        if(n&1)
```

```
        {
```

```
            ans=multimod(ans,a,p);
```

```
        }
```

```
    }
```

```

        a=multimod(a,a,p);

        n=n>>1;
    }

    return ans;
}

INT euler(INT n)
{
    INT i;

    INT total=n;

    for(i=0;i<=top&&prime[i]*prime[i]<=n;i++)
    {
        if(n%prime[i]==0)
        {
            total=total/prime[i]*(prime[i]-1);

            while(n%prime[i]==0)
                n/=prime[i];
        }
    }

    if(n!=1)
        total=total/n*(n-1);

    return total;
}

INT solve(INT m)
{
    INT phi=euler(m);

    INT i;

    INT min=0;

    for(i=1;i*i<=phi;i++)
    {
        if(phi%i==0)

```

```

    {
        INT temp=quickmod(10,i,m);

        if(temp==1)
        {
            min=i;

            break;
        }

        temp=quickmod(10,phi/i,m);

        if(temp==1)
        {
            if(min==0)

                min=phi/i;

            else
            {
                if(min>phi/i)

                    min=phi/i;
            }
        }
    }

    if(min==0)

        min=phi;

    return min;
}

int main(void)
{
    init();

    INT l;

    int cas=1;

    while(scanf("%lld",&l)==1)
    {

```

```
    if(l==0)

        break;

    printf("Case %d: ",cas++); //先开始把这句话打到最前面，WA 到吐！！真是。。

    if(l==1)

    {

        puts("1");

        continue;

    }

    if((l%16==0) || (l%5==0))

    {

        puts("0");

        continue;

    }

    INT t=gcd(l,8LL);

    l/=t;

    l*=9;

    printf("%lld\n",solve(l));

}

return 0;

}
```