> 1. 定义$Fibonacci$数列如下：$F(0)=0$，$F(1)=1$，且对于$n\geq 2$，$F(n)=F(n-1)+F(n-2)$。所以，该数列是：$0,1,1,2,3,5,8,13,21,\ldots$ 。如何能快速地求出$F(n)$呢？很幸运，我们有以下等式：
>
> $$\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}^n=\begin{bmatrix}F(n+1) & F(n)\\ F(n) & F(n-1)\end{bmatrix}$$
>
> 虽然，看上去该算法需要一次矩阵的指数运算，但是借助快速指数运算的方法，这里可以产生一个快速求解$F(n)$的算法。请给出算法，并编程实现。

解：

令

$$x=\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix},res=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$$

根据快速指数运算方法：将n二进制展开，假设有k个bit，将x^n变形为：
$x^n=\prod_{i=0}^{k-1}x^{n_i^{2^i}}$

不过此处乘法做的是矩阵乘法。

代码如下：

```
//x= [ a_1  b_1 ]   res = [ a_2  b_2 ]
//   [ c_1  d_1 ]         [ c_2  d_2 ]
int Fibonacci(int n)
{
    int a_1 = 1, b_1 = 1, c_1 = 1, d_1 = 0;
    int a_2 = 1, b_2 = 0, c_2 = 1, d_2 = 0;
    while (n > 0)
    {
        if ((n & 1) == 1)
        {
            //res = (res * x)
            int temp_a = a_2, temp_b = b_2, temp_c = c_2, temp_d = d_2;
            a_2 = temp_a * a_1 + temp_b * c_1;
            b_2 = temp_a * b_1 + temp_b * d_1;
            c_2 = temp_c * a_1 + temp_d * c_1;
            d_2 = temp_c * b_1 + temp_d * d_1;
        }
```

```
        n /= 2; //右移1bit

        //x = x * x
        int temp_a = a_1, temp_b = b_1, temp_c = c_1, temp_d = d_1;
        a_1 = temp_a * temp_a + temp_b * temp_c;
        b_1 = temp_a * temp_b + temp_b * temp_d;
        c_1 = temp_a * temp_c + temp_c * temp_d;
        d_1 = temp_b * temp_c + temp_d * temp_d;
    }
    //return res
    return b_2;
}
```

> 2. 给定任意正整数n，n的所有因子分别记为$d_0, d_1, \ldots, d_r$，其中包括1和n。记函数：
>
> $$F(n) = \phi(d_0) + \phi(d_1) + \cdots + \phi(d_r)$$
>
> 现在需要证明$F(n) = n$。为完成这个任务，请依次完成以下小任务：
>
> (a).证明，对任意素数p，$F(p) = p$。
>
> (b).证明，对任意素数p，$F(p^2) = p^2$
>
> (c).证明，对任意素数p，$F(p^k) = p^k$，k是任意正整数
>
> (d).证明，对任意素数p和q，$F(pq) = pq$
>
> (e).证明函数F是积性函数，即对任意的正整数m和n，如果$gcd(m, n) = 1$，则$F(mn) = F(m)F(n)$。
>
> (f).最后，完成证明$F(n) = n$的任务。

证明：

(a). 对于任意素数p，其只有两个因子,分别是1和p本身。

$$F(p) = \phi(1) + \phi(p) = 1 + (p - 1) = p$$

(b). 对于任意素数p，

$$F(p^2)=\phi(1)+\phi(p)+\phi(p^2)=1+(p-1)+(p^2-p)=p^2$$

(c). 对于任意素数p，

$$F(p^k)=\phi(1)+\sum_{i=1}^{k}\phi(p^i) \tag{*}$$

对于每个$\phi(p^i)$，$i\in\{1,2,\dots k\}$，都有：$\phi(p^i)=p^i-p^{i-1}$

$$\begin{gathered}p^k-p^{k-1}=\phi(p^k)\\ p^{k-1}-p^{k-2}=\phi(p^{k-1})\\ \dots\\ p^2-p=\phi(p^2)\\ p-1=\phi(p)\end{gathered}$$

累加得：

$$p^k-1=\sum_{i=1}^{k}\phi(p^i)$$

将上式代入(*)：

$$F(p^k)=1+(p^k-1)=p^k$$

(d). 对于任意素数p和q，

$$F(pq)=\phi(p)+\phi(q)+\phi(1)+\phi(pq) \tag{**}$$

其中:$\phi(pq)=\phi(p)\phi(q)=(p-1)(q-1)$，$\phi(q)=q-1$，$\phi(p)=p-1$，代入(**)得：

$$F(pq)=(p-1)+(q-1)+1+(p-1)(q-1)=pq$$

(e). 对任意正整数m和n,因为$gcd(m,n)=1$，故对于m,n的因子都各自互素。

记m的所有因子分别为$d_{0_m},d_{1_m},\dots,d_{r_m}$，其中包括1和$m$。共有$(r-1)$个因子,有：

$$F(m)=\phi(d_{0_m})+\phi(d_{1_m})+\dots+\phi(d_{r_m})=\sum_{i=0}^{r}d_{i_m} \tag{1}$$

记n的所有因子分别为$d_{0_n}, d_{1_n}, \dots, d_{t_n}$，其中包括1和$n$。共有$(t-1)$个因子，有：

$$F(n) = \phi(d_{0_n}) + \phi(d_{1_n}) + \cdots + \phi(d_{r_n}) = \sum_{j=0}^{t} d_{j_n} \tag{2}$$

则mn因子为$d_{i_m} d_{j_n}$, 其中$i \in \{0, 1, \dots, r\}, j \in \{0, 1, \dots, t\}$。共有$(r-1)(t-1)$个因子。

又因每个d_{i_m}与d_{j_n}互素，有$\phi(d_{i_m} d_{j_n}) = \phi(d_{i_m})\phi(d_{j_n})$，故有：

$$F(mn) = \sum_{j=0}^{t} \sum_{i=0}^{r} \phi(d_{i_m} d_{j_n}) = \sum_{i=0}^{r} \phi(d_{i_m}) \sum_{j=0}^{t} \phi(d_{j_n}) = F(m)F(n)$$

(f). 将正整数n表达为素数指数的乘积，即$n = p_1^{a_1} p_2^{a_2} \dots p_k^{a_k}$，有：

$$F(n) = F(p_1^{a_1} p_2^{a_2} \dots p_k^{a_k})$$

根据$(d). F(pq) = pq$ 和 $(e). F(mn) = F(m)F(n)$

将$p_2^{a_2} \dots p_k^{a_k}$视为整体，则有：

$$F(n) = F(p_1^{a_1}(p_2^{a_2} \dots p_k^{a_k})) = F(p_1^{a_1})F(p_2^{a_2} \dots p_k^{a_k})$$

将$p_3^{a_3} \dots p_k^{a_k}$视为整体，则有：

$$F(p_2^{a_2}(p_3^{a_3} \dots p_k^{a_k})) = F(p_2^{a_2})F(p_3^{a_3} \dots p_k^{a_k})$$

以此类推：

$$F(n) = F(p_1^{a_1} p_2^{a_2} \dots p_k^{a_k}) = F(p_1^{a_1})F(p_2^{a_2}) \dots F(p_k^{a_k})$$

根据$(c). F(p^k) = p^k$

$$F(n) = F(p_1^{a_1})F(p_2^{a_2}) \dots F(p_k^{a_k}) = p_1^{a_1} p_2^{a_2} \dots p_k^{a_k}$$

而 $p_1^{a_1} p_2^{a_2} \dots p_k^{a_k} = n$，故

$$F(n) = n$$