实验四 公钥密码算法 RSA

一、 实验目的

通过实际编程了解公钥密码算法 RSA 的加密和解密过程，加深对公钥密码算法的了解和使用。

二、 实验原理

序列密码和分组密码算法都要求通信双方通过交换密钥实现使用同一个密钥，这在密钥的管理、发布和安全性方面存在很多问题，而公钥密码算法解决了这个问题。

公钥密码算法是指一个加密系统的加密密钥和解密密钥是不同的，或者说不能用其中一个推导出另一个。在公钥密码算法的两个密钥中，一个是用于加密的密钥，它是可以公开的，称为公钥；另一个是用于解密的密钥，是保密的，称为私钥。公钥密码算法解决了对称密码体制中密钥管理的难题，并提供了对信息发送人的身份进行验证的手段，是现代密码学最重要的发明。

RSA 密码体制是目前为止最成功的公钥密码算法，它是在 1977 年由 Rivest、Shamir 和 Adleman 提出的第一个比较完善的公钥密码算法。它的安全性是建立在“大数分解和素性检测”这个数论难题的基础上，即将两个大素数相乘在计算上容易实现，而将该乘积分解为两个大素数因子的计算量相当大。虽然它的安全性还未能得到理论证明，但经过 20 多年的密码分析和攻击，迄今仍然被实践证明是安全的。

RSA 算法描述如下：

1、公钥

选择两个不同的大素数 p 和 q，n 是二者的乘积，即 n＝pq，使

$$\varphi(n) = (p-1)(q-1)$$

$\varphi(n)$ 为欧拉函数。随机选取正整数 e，使其满足 $(e,\varphi(n)) = 1$，即 e 和 $\varphi(n)$ 互素，则将（n，e）作为公钥。

2、私钥

求出正数 d，使其满足 $e \times d \equiv 1 \bmod \varphi(n)$，则将（n，d）作为私钥。

3、密算法

对于明文 m，由 $c \equiv m^e \bmod n$，得到密文 c。

4、解密算法

对于密文 c，由 $m \equiv c^d \bmod n$，得到明文 m。

如果攻击者获得了 n、e 和密文 c，为了破解密文必须计算出私钥 d，为此需要先分解 n。当 n 的长度为 1024 比特时，在目前还是安全的，但从因式分解技术的发展来看，1024 比特并不能保证长期的安全性。为了保证安全性，要求在一般的商业应用中使用 1024 比特的长度，在更高级别的使用场合，要求使用 2048 比特长度。

三、 实验环境

运行 Windows 操作系统的 PC 机，具有 VC 等语言编译环境

四、 实验内容和步骤，

1、为了加深对 RSA 算法的了解，根据已知参数：p＝3，q＝11，m＝2，手工计算公钥和私钥，并对明文 m 进行加密，然后对密文进行解密。

2、编写一个程序，用于生成 512 比特的素数。

3．利用 2 中程序生成的素数，构建一个 n 的长度为 1024 比特的 RSA 算法，利用该算法实现对明文的加密和解密。

4、在附件中还给出了一个可以进行 RSA 加密和解密的对话框程序 RSATool，运行这个程序加密一段文字，了解 RSA 算法原理。

五、 实验报告和要求

1、对实验步骤 2，写出生成素数的原理，包括随机数的生成原理和素性检测的内容，并给出程序框图；

2、对实验步骤 3，要求分别实现加密和解密两个功能，并分别给出程序框图。