这道题涉及到了一个使用较为普遍的非对称加密算法——RSA。

所以做题之前，有必要简短快速地回顾一下 RSA 非对称加密的过程。（PS：如果事先了解过，建议跳过这个部分）

一个上千位的大整数要分解为几个质数相乘，这在现有数学基础上是十分困难的，而 RSA 就是利用了这一点。

举个例子，24=2*2*2*3，这很简单。

那么，如果要分解的数字是 2808610417 呢？

答案是：2808610417=61381*45757。这已经很复杂了，但我们常用来加密的质数一般是 1024bit 或 2048bit，那就更加困难了。

说了这么多，RSA 到底是如何加密的？

1. **选择两个大素数：**
    - 选择两个不同的大素数 p 和 q。这两个素数需要保密，因为它们是生成密钥的基础。
2. **计算 n 和 $\phi(n)$：**
    - 计算 $n=p\times q$。
    - 计算欧拉函数 $\phi(n)=(p-1)\times(q-1)$。
3. **选择公钥指数 e：**
    - 选择一个整数 e，满足 $1<e<\phi(n)$ 且 $\gcd(e,\phi(n))=1$。通常选择 $e=65537$，因为它是一个常用的素数，且计算

效率高。

4. **计算私钥指数** d：
   - 计算 d，使得 $d \times e \equiv 1 (\mathrm{mod} \phi(n))$。这可以通过扩展欧几里得算法来完成。这里便是计算乘法逆元。
5. **生成密钥对**：
   - 公钥：(e, n)
   - 私钥：(d, n)
6. **加密**：
   - 对明文 m 进行加密，计算密文 c：

$$c = m^e \mod n$$

7. **解密**：
   - 对密文 c 进行解密，恢复明文 m：

$$m = c^d \mod n$$

所以大部分 RSA 加密的问题在于找到私钥 d。

而这道 babyrsa 中的 n 就是一个质数，那么根据欧拉函数的性质，

phi(n)=n-1，然后就可以计算出 d，进而解密。