CS 173, Spring 2013
Handout on RSA for Honors Homework

To do the honors homework on RSA encryption, you should read this handout and also pp. 131-134 from Liebeck, *A Concise Introduction to Pure Mathematics*, 2nd edition, Chapman and Hall, 2006.

1 Extended Euclidean algorithm

Suppose that we have two integers p and q, whose gcd is g. Then the equation $g = px+qy$ has integer solutions. We can use an extension of the Euclidean algorithm to find one solution.

Remember, in the Euclidean algorithm, we take our original integers p and q (assume $p \geq q$) and make a sequence of integers $p = r_1, q = r_2, r_3, r_4, \ldots r_n$ such that

$$\gcd(p, q) = \gcd(r_1, r_2) = \gcd(r_2, r_3) = \gcd(r_3, r_4) \ldots = \gcd(r_{n-1}, r_n)$$

Each integer in this sequence is produced by dividing the two previous integers and taking the remainder. This gives us a series of integer division equations of the form $r_{k-1} = mr_k + r_{k+1}$. In each of these, we could solve for r_{k+1}: $r_{k+1} = r_{k-1} - mr_k$.

For example, in computing the gcd of 5817 and 1428 (which is 21), we find that

$$\begin{aligned}
5817 &= 4 \cdot 1428 + 105 \\
1428 &= 13 \cdot 105 + 63 \\
105 &= 1 \cdot 63 + 42 \\
63 &= 1 \cdot 42 + 21
\end{aligned}$$

So

$$\begin{aligned}
105 &= 5817 - 4 \cdot 1428 \\
63 &= 1428 - 13 \cdot 105 \\
42 &= 105 - 63 \\
21 &= 63 - 42
\end{aligned}$$

Now, to solve the equation $21 = 5817x + 1428y$, we use the above equations in reverse order. Start with the bottom equation, which expresses the gcd in terms of the smallest two elements in the sequence:

$21 = 63 - 42$

Get rid of the smaller number on the righthand side by substituting in the righthand side of the previous equation:

$21 = 63 - (105 - 63) = 2 \cdot 63 - 105$

Do this again, to get rid of 63:

$21 = 2 \cdot (1428 - 13 \cdot 105) - 105 = 2 \cdot 1428 - 27 \cdot 105$

And again to remove 105:

$21 = 2 \cdot 1428 - 27 \cdot (5817 - 4 \cdot 1428) = -27 \cdot 5817 + 110 \cdot 1428$

So our final result: $21 = -27 \cdot 5817 + 110 \cdot 1428$

2 Successive Squares

Suppose that we want to compute a number like $6^{82} \bmod 13$. Since the answer is between 0 and 12, it seems inefficient to get it by computing a really huge intermediate quantity like 6^{82}. And, in fact, it's possible to compute it easily by hand.

To see how the trick works, let's represent the exponent as the sum of powers of two (as in base-2 numbers). $82 = 64 + 16 + 2$. So

$$6^{82} = 6^{64} \cdot 6^{16} \cdot 6^{2}$$

We can raise 6 to a power of two by successive squaring. Recall that if $a \equiv b \pmod{m}$ then $a^n \equiv b^n \pmod{m}$, for any natural number n. So, each time we square, we can convert the result to a handy (i.e. small) integer that's equivalent mod 13.

In this case

$$
\begin{aligned}
6^2 &= (-3) \pmod{13} \\
6^4 &= 9 \pmod{13} \\
6^8 &= 3 \pmod{13} \\
6^{16} &= 9 \pmod{13} \\
6^{32} &= 3 \pmod{13} \\
6^{64} &= 9 \pmod{13}
\end{aligned}
$$

So then

$$6^{82} = 6^{64} \cdot 6^{16} \cdot 6^2 \equiv 9 \cdot 9 \cdot (-3) \pmod{13}$$

But then $9 \cdot -3 = -27 \equiv -1 \pmod{13}$. So $9 \cdot 9 \cdot -3$ is congruent to $9 \cdot (-1)$, which is congruent to 4, mod 13. So $6^{82} \equiv 4 \pmod{13}$.

3 RSA "Encryption"

The RSA function was proposed as a "public-key encryption" scheme in 1977. However, the original RSA scheme, or "textbook RSA" as it is now known, is by itself not a sufficiently secure encryption scheme (since, for instance, it produces the same ciphertext each time the same message is encoded using the same key – which would let an eavesdropper infer that a message is being sent again, even though she won't necessarily learn its contents). But variants which do rely on the RSA (along with some random padding) form the basis of a popular encryption standard today. Below we discuss only the original (textbook) RSA encoding and decoding schemes.

When Liebeck (page 133) explains how to decode a message, you don't really have to understand all of the first couple paragraphs. The short version is:

Decoding and encoding are done the same way. To encode x, compute $y = x^e \bmod N$ to decode y, compute $x = x^d \bmod N$. The trick is to find the d that goes with a particular e.

Suppose you know N and e and p and q. Suppose we set $z = (p-1)(q-1)$. For reasons that you don't have to understand (that's the reference to proposition 15.3 in Liebeck), you can find d by solving the equation

$$1 = de + kz$$

You can do this using the method in section 1 above. (Thus RSA can be broken if the prime factorization of N can be efficiently computed.)

For Liebeck's example (paragraph 2), $e = 11$ and $z = 2160$. So he sets up the equation:

$$1 = d \cdot 11 + k \cdot 2160$$

A solution to it is:

$$1 = 1571 \cdot 11 - 8 \cdot 2160$$

So 1571 is a suitable value for d.

Sometimes if you follow this procedure, you end up with a negative value for the coefficient of e. E.g.

$$1 = mz - fe$$

Where all the variables are positive. $-f$ is no good as a value for d.

Notice that this equation has lots of solutions. In particular, another one is

$$1 = (m - e)z + (z - f)e$$