1. Consider the RSA encryption scheme with public keys $n = 55$ and $e = 7$.

   (a) [4 points] Encipher the plaintext $M = 19$. Use the binary exponentiation algorithm and show your work.

   Solution:

   To encipher M we compute

$$C \equiv M^e \equiv 19^7 \pmod{55}.$$

   To use binary exponentiation, we need the binary expansion of $e = 7$. Since $7 = 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$ we get $b_0 = 1$, $b_1 = 1$, and $b_2 = 1$. Then

$$\begin{aligned} r_0 &\equiv 19^{b_0} \equiv 19 \pmod{55}, \\ r_1 &\equiv (r_0)^2 19^{b_1} \equiv (361)(19) \equiv (31)(19) \equiv 589 \equiv 39 \pmod{55}, \\ r_2 &\equiv (r_1)^2 19^{b_2} \equiv (1521)(19) \equiv (36)(19) \equiv 684 \equiv 24 \pmod{55}. \end{aligned}$$

   Thus, $19^7 \equiv r_2 \equiv 24 \pmod{55}$, and $C = 24$.

   (b) [4 points] Break the cipher by finding p, q, and d.

   Solution:

   To find p and q we factor $n = 55 = 5 \times 11$, yielding $p = 5$, $q = 11$. Thus $\phi(n) = (p-1)(q-1) = 40$, and we compute d by solving the linear congruence $ed \equiv 1 \pmod{\phi(n)}$ or

$$7d \equiv 1 \pmod{40}.$$

   We use the Extended Euclidean Algorithm to solve the associated linear Diophantine equation

$$7d + 40k = 1$$

   for d. We first compute the sequence of q's using the Euclidean algorithm:

$$\begin{aligned} 7 &= 0(40) + 7, & q_0 &= 0, \\ 40 &= 5(7) + 5, & q_1 &= 5, \\ 7 &= 1(5) + 2, & q_2 &= 1, \\ 5 &= 2(2) + 1, & q_3 &= 2, \\ 2 &= 2(1) + 0, & q_4 &= 2. \end{aligned}$$

   Thus $n = 4$, and we compute $d = (-1)^{n-1} B_{n-1}$ where B_{n-1} is computed using the recurrence defined by $B_{-2} = 1$, $B_{-1} = 0$ and

$$B_k = q_k B_{k-1} + B_{k-2}, \quad k = 0, \ldots, n.$$

   We obtain

i	-2	-1	0	1	2	3
q_i	–	–	0	5	1	2
B_i	1	0	1	5	6	17

yielding $d \equiv (-1)^3 17 \equiv -17 \equiv 23 \pmod{40}$. To check, observe that $7(23) = 161 \equiv 1 \pmod{40}$.

(c) [4 points] Decipher the ciphertext $C = 35$. Use the binary exponentiation algorithm and show your work.

Solution:

To decrypt C we compute

$$M \equiv C^d \equiv 35^{23} \pmod{55}.$$

To use binary exponentiation, we need the binary expansion of $d = 23$. Since $23 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$ we get $b_0 = 1$, $b_1 = 0$, $b_2 = 1$, $b_3 = 1$, and $b_4 = 1$. Then

$$\begin{aligned}
r_0 &\equiv 35^{b_0} \equiv 35 \pmod{55},\\
r_1 &\equiv (r_0)^2 35^{b_1} \equiv (1225) \equiv 15 \pmod{55},\\
r_2 &\equiv (r_1)^2 35^{b_2} \equiv (225)(35) \equiv (5)(35) \equiv 175 \equiv 10 \pmod{55},\\
r_3 &\equiv (r_2)^2 35^{b_3} \equiv (100)(35) \equiv (45)(35) \equiv 1575 \equiv 35 \pmod{55},\\
r_4 &\equiv (r_3)^2 35^{b_4} \equiv (1225)(35) \equiv (15)(35) \equiv 525 \equiv 30 \pmod{55}.
\end{aligned}$$

Thus $35^{23} \equiv r_4 \equiv 30 \pmod{55}$, and $M = 30$.

2. [6 points] It is obvious that if one can factor an RSA modulus $n = pq$, i.e. one knows the prime factors p, q of n, then one can compute $\phi(n) = (p-1)(q-1)$. Prove the converse, i.e. if both n and $\phi(n)$ are known, then p and q can be found without factoring n.

Solution:

We have

$$\phi(n) = (p-1)(q-1) = pq - p - q + 1 = n - p - \frac{n}{p} + 1,$$

so $p\phi(n) = pn - p^2 - n + p$, or equivalently,

$$p^2 + (\phi(n) - n - 1)p + n = 0.$$

The equation $x^2 + (\phi(n) - n - 1)x + n = 0$ is a quadratic equation with known coefficients, and we can solve it using the well-known formula

$$x = -\frac{\phi(n) - n - 1}{2} \pm \sqrt{\left(\frac{\phi(n) - n - 1}{2}\right)^2 - n}.$$

The equation will have two solutions: p (for the "+" sign) and q (for the "−" sign).

3. This problem describes a "difference of squares" attack on RSA. Suppose two RSA primes p and q $(q > p)$ are very close to one another, i.e. $q = p + \delta$ where $\delta \in \mathbb{N}$ is small (i.e. small enough that it is feasible to try all possible values 1, 2, 3, ... for δ; for example, we could have $\delta \approx \log p$). Note that in this case, $p + q$ is only slightly larger than $\sqrt{n}$.

   (a) [5 points] Using the identity

   $$\left(\frac{q+p}{2}\right)^2 = n + \left(\frac{q-p}{2}\right)^2,$$

   describe an algorithm to recover $p + q$.

   Solution:

   We have $(q+p)^2 = 4n+\delta^2$. Note that since p and q are both odd, their difference δ must be even. So all we need to do is check whether for $i = 2, 4, 6, \ldots$, the quantity $4n + i^2$ is a square; that is, take $\sqrt{4n + i^2}$ and check whether it is an integer. This requires at most $\delta/2$ trials. Every value of i such that $4n + i^2$ is a square gives us a candidate for $p + q$ (in fact, there will be a unique such value of i by unique factorization and part (b)).

   (b) [3 points] Using the technique of part (a), describe a way to recover p and q efficiently without factoring n.

   Solution:

   We know n and (from part (a)) $p + q$. Now $\phi(n) = (p - 1)(q - 1) = n - (p + q) + 1$, so we can find $\phi(n)$. By Problem 1, we can now easily derive p and q.

   (c) [2 points] Explain why $n = 23614161161$ is a particularly bad choice as an RSA modulus (apart from the fact that it's too small to guarantee a decent level of security).

   Solution:

   The prime factorization of 23614161161 is $n = pq$ with $p = 153649$ and $q = 153689$. We have $p - q = 40$, so we can factor n after at most 20 trials, using the technique in (a).

   Note that this problem raises an important practical point: when choosing RSA primes, make sure that they are not too close together!

4. After the discovery of RSA, several writers suggested using it with a small encryption exponent e (for example, $e = 2, 3$). Show why using such a small exponent is insecure in the following scenarios:

   (a) [8 points] Two people send the same message M to two different receivers. A different modulus is used for each transmission, but $e = 2$ for both.

   Solution:

   Suppose Alice is sending M to Bob using Bob's public key $(n_B, e_B = 2)$ and also to Carol using Carol's public key $(n_C, e_C = 2)$. Thus, Alice sends $C_B \equiv M^2 \pmod{n_B}$ to Bob and $C_C \equiv M^2 \pmod{n_C}$ to Carol.

   An adversary with C_B and C_C who knows that these two ciphertexts are encryptions of the same message can compute M as follows. First he computes $\gcd(n_B, n_C)$. If by some miracle this give an answer other than 1, then both moduli can be factored and both secret keys found. Otherwise, he can use the Chinese Remainder Theorem to solve

$$X \equiv \begin{cases} C_B \pmod{n_B} \\ C_C \pmod{n_C} \end{cases}$$

for $X \pmod{n_B n_C}$, $0 < X < n_B n_C$. Then $X \equiv M^2 \pmod{n_B}$ and $X \equiv M^2 \pmod{n_C}$, so (since n_B and n_C are coprime) $X \equiv M^2 \pmod{n_B n_C}$. But since $0 < M < n_B$ and $0 < M < n_C$, we have $0 < M^2 < n_B n_C$, so $X = M^2$ and hence $M = \sqrt{X}$.

(b) [8 points] Two different messages which differ by only a few characters (the adversary can deduce the position of these characters) are sent under the same key. Here, $e = 2$ and n is the same for both messages.

Solution:

Suppose messages M_1 and M_2 are sent using the same public key $e = 2$ and the same modulus n, i.e. $C_1 \equiv M_1^2 \pmod{n}$ and $C_2 \equiv M^2 \pmod{n}$. Now $M_1 = M_2 + r$ for some nonzero $r \in \mathbb{Z}$. Since by assumption, the adversary can deduce the positions of the characters where M_1 and M_2 differ, there are only a small number of possible values for r, and the adversary can explicitly determine all of them and try them all on the procedure given below.

Do the following for each candidate r: if by some miracle $\gcd(n, r) > 1$, then n can be factored, so assume that n and r are coprime. Then

$$C_1 \equiv M_1^2 \equiv (M_2 + r)^2 \equiv M_2^2 + 2M_2 r + r^2 \pmod{n},$$

and since $C_2 \equiv M_2^2 \pmod{n}$, we have $C_1 \equiv C_2 + 2M_2 r + r^2 \pmod{n}$. Thus,

$$2rM_2 \equiv C_1 - C_2 - r^2 \pmod{n}.$$

Since n is odd and $\gcd(r, n) = 1$, the adversary can easily compute $s \in \mathbb{Z}$, $0 < s < n$ such that $2rs \equiv 1 \pmod{n}$. Then $M \equiv s(C_1 - C_2 - r^2) \pmod{n}$, $0 < M < n$, is found.

5. [10 points] Rabin's public-key encryption scheme enciphers a message M as

$$C \equiv M(M + b) \pmod{n}, \quad (0 \le C < n)$$

where b and n are public and $n = pq$ for secret primes p and q. Give a deciphering algorithm for the case where $p + 1$ and $q + 1$ are divisible by 4.

Hint 1: Compute d such that $2d \equiv b \pmod{n}$. Then

$$C + d^2 \equiv (M + d)^2 \pmod{n}.$$

Hint 2: If $x^2 \equiv a \pmod{p}$ and p is a prime such that $p \equiv 1 \pmod{4}$, then

$$x \equiv \pm a^{(p+1)/4} \pmod{p}$$

are the two square roots of $a \pmod{p}$ (you need not prove this).

Solution:

Compute d such that $2d \equiv b \pmod{n}$. Then $C + d^2 \equiv (M+d)^2 \pmod{n}$. Set $a = C + d^2$. We need to find x such that $x^2 \equiv a \pmod{n}$, i.e., we need to find a square root of $a \pmod{n}$.

Compute $r_p \equiv a^{(p+1)/4} \pmod{p}$ and $r_q \equiv a^{(q+1)/4} \pmod{q}$. Then by the second hint $r_p^2 \equiv (-r_p)^2 \equiv a \pmod{p}$ and $r_q^2 \equiv (-r_q)^2 \equiv a \pmod{q}$, i.e., r_p and $-r_p$ are square roots of $a \pmod{p}$ and r_q and $-r_q$ are square roots of $a \pmod{q}$.

Select one square root of $a \pmod{p}$ (take r_p) and one modulo q (take r_q). We use the Chinese Remainder Theorem to compute x such that $x \equiv r_p \pmod{p}$ and $x \equiv r_q \pmod{q}$. Note that $x^2 \equiv a \pmod{n}$, since $x^2 \equiv a \pmod{p}$, $x^2 \equiv a \pmod{q}$ and $n = pq$. Since there are two choices for the root modulo p and two for the root modulo q, we obtain four distinct square roots of a modulo n by this method.

Let x_1, x_2, x_3, x_4 be the four square roots of $a \equiv C + d^2 \pmod{n}$. Then since $x_i \equiv M + d \pmod{n}$ for $i = 1, 2, 3$, or 4, M is equivalent to one of $x_1 - d, x_2 - d, x_3 - d, x_4 - d$ modulo n. One of these four values is the correct message.

There is no way to algorithmically determine which of the four possible messages was sent. If the message is English or some other language that has fixed redundancy characteristics, the correct message can easily be determined by examining all four possibilities. Another solution is to append a small bit of fixed text to a message before encrypting. The decryption which contains the same fixed text at the end is taken as the correct message.

(Aside: Breaking Rabin's scheme is *provably equivalent* to factoring, unlike RSA. This means that if one has a fast algorithm for breaking Rabin's scheme, then that algorithm can be used to factor n quickly.)

6. [6 points] Let $n = pq$ for distinct primes p and q. Given a, $0 < a < n$, let x and y, $0 < x, y < n$, be square roots of a modulo n, so

$$x^2 \equiv a \pmod{n} \quad \text{and} \quad y^2 \equiv a \pmod{n}.$$

Show that $\gcd(x+y, n) = p$ or q if $y \neq x$ and $y \neq n - x$, i.e., finding such x and y allows one to factor n.

Solution:

Since $x^2 \equiv a \pmod{n}$ and $y^2 \equiv a \pmod{n}$, we have $x^2 \equiv y^2 \pmod{n}$. Hence $n = pq$ divides $x^2 - y^2 = (x-y)(x+y)$.

Since $x, y < n$ it follows that $|x - y| < n$, and since $x \neq y$ implies $x - y \neq 0$, we see that n cannot divide $x - y$. Since $0 < x, y < n$, it follows that $0 < x + y < 2n$. Also, $y \neq n - x$ implies $x + y \neq n$, and hence n cannot divide $x + y$. Thus, since n does not divide either $x - y$ or $x + y$, the fact that n divides $(x+y)(x-y)$ implies that either p or q must divide $x + y$ yielding $\gcd(n, x+y) = p$ or q.