**Problem 1.** The point of this problem is to show that Fermat’s test for compositeness (i.e., the use of Fermat’s little theorem) is theoretically usable, even for Carmichael numbers, but that in the real world it is hopeless for Carmichael numbers.

Fix a Carmichael number \( n \). Recall that this means that \( n \) is composite with the property that \( a^{n-1} \equiv 1 \mod n \) for all integers \( a \) coprime to \( n \).

(a) If \( \gcd(a,n) > 1 \) then \( a^{n-1} \not\equiv 1 \mod n \). This is the good news; there are some \( a \)’s that detect \( n \)’s compositeness.

(b) To run Fermat’s test, you choose an \( a \) in the range \( 1, \ldots, n-1 \) and see if \( a^{n-1} \equiv 1 \mod n \). By (a) you know that the test detects \( n \)’s compositeness if and only if \( (a,n) > 1 \). Suppose our Carmichael number \( n \) is the product of three distinct primes \( p, q \) and \( r \). Write down the probability that a random \( a \) from the range \( 1, \ldots, n-1 \) detects \( n \)’s compositeness. (The answer will involve \( p \) and \( q \) and \( r \). It turns out that a Carmichael number cannot be the product of just two primes. That’s why I assumed three primes.)

(c) Continuing as in (b), suppose that \( p, q \) and \( r \) are all large, say \( \approx 10^{30} \). Estimate the probability that a single randomly chosen \( a \) detects \( n \)’s compositeness.

(d) Continuing as in (c), imagine that you keep choosing \( a \) randomly from \( 1, \ldots, n-1 \), until you detect \( n \)’s compositeness. Say you can test a single value of \( a \) in a billionth of a second. How many years do you expect to wait before you detect \( n \)’s compositeness? Compare this with the age of the universe.

(Remark: a 30-digit prime is tiny for modern cryptographic applications. The NSA documents revealed by Ed Snowden recommend using 4096-bit keys, which means that the primes involved should be 616 decimal digits or so. Using 30-digit primes would get you laughed at in computer security circles.)

**Problem 2.** This problem is the good news to counter the previous problem’s bad news. It proves that Fermat’s little theorem is very effective, as long as the number \( n \) to be tested for compositeness is not a Carmichael number. The goal is to show that at least half of the integers \( a = 1, \ldots, n-1 \mod n \) detect the compositeness using Fermat’s Little Theorem.

(a) All the \( a \) with \( \gcd(a,n) > 1 \) detect \( n \)’s compositeness. So it suffices to show that at least half of the \( a \)’s that are coprime to \( n \) detect \( n \)’s compositeness.

(b) Write \( E \) for the number of solutions to \( x^{n-1} \equiv 1 \). Then for every \( b \) coprime to \( n \), the number of solutions to \( x^{n-1} \equiv b \mod n \) is either 0 or \( E \).

(c) Assuming that \( n \) is not Carmichael, so that there is some \( a \) coprime to \( n \) such that \( a^{n-1} \not\equiv 1 \mod n \), use (b) to show that at least half of the \( a \)’s that are coprime to \( n \) detect \( n \)’s compositeness.

(d) How many randomly chosen \( a \)’s do you expect to try before detecting \( n \)’s compositeness?

**Problem 3.** If \( p \) and \( q \) are distinct primes then \( p^{n-1} + q^{n-1} \equiv 1 \mod pq \).

**Caution:** at the bottom of p. 323 in the text there is an extremely confusing typo. The right side should be \( C^d \) not \( P^d \).

**Problem 4.** Encrypt the message \( 1204 \ 1106 \ 2106 \ 0038 \ 0607 \) using RSA with modulus \( n = 2669 \) and encrypting exponent \( e = 3 \). (This modulus is laughably small.
for real-world use. But this does take you through the mechanics of RSA encryption. Encoding just plain numbers probably seems boring, but actually converting to/from letters

*Problem 5.* Suppose an incompetent spy has a long list of numbers, say 2, . . . , 1 000 000, representing common spy words like “rendevous” or “kompromat” and so forth. He uses a huge modulus $n$, say a 100-digit number, but a small encrypting exponent, say $e \leq 16$. Explain why someone listening to his communications can read his messages, without having to factor $n$.

*Problem 6.* Read the Wikipedia pages on “Frequency Analysis” and “Bigram”, and also read (lightly) the book’s discussion about how to convert letters to numbers for using RSA crypto. Nothing to turn in, but I hope you will see that the book’s approach offers very little security, even though RSA itself is very secure. (In the book’s defense, this difficulty arises from choosing small numbers, typically 4-digit, as the moduli for RSA crypto. This is unrealistic but reasonable for teaching purposes.)