Tutorial 5
Sequences, Modulo Arithmetic, LCM

1 Discussion questions

Discussion questions are meant for discussion on the IVLE Forum. You may try them on your own or discuss them with your classmates. No answers will be provided by us.

D1. The Triangle numbers $\triangle_n$ are: $0, 1, 3, 6, 10, 15, 21, \ldots$. Write the recurrence relation and initial conditions for this sequence. Do the same for the Square numbers, $\square_n$: $0, 1, 4, 9, 16, 25, \ldots$.

D2. Let's sum the first n Triangle numbers, that is, define $T_n = \sum_{k=0}^{n} \triangle_k$, for $n \in \mathbb{N}$. Determine $T_0, T_1, \ldots, T_8$. Can you find an explicit formula for T_n?

D3. In the lecture on Sequences, a sequence was shown as follows:
$1, 3, 5, 7, 217341, \ldots$.
Find an explicit formula $f(n), \forall n \in \mathbb{Z}^+$ that generates this sequence.

2 Tutorial questions

Q1. The Lucas numbers, L_n, are given by the recurrence:

$$L_n = \begin{cases} 2, & \text{if } n = 0, \\ 1, & \text{if } n = 1, \\ L_{n-1} + L_{n-2}, & \text{for } n > 1. \end{cases}$$

They are similar to the Fibonacci numbers F_n except for the initial conditions.

(a) Work out by hand the Lucas numbers $L_0, L_1, \ldots, L_9$.

(b) Using the recurrence relation alone (ie. not the explicit formula), prove by Mathematical Induction that $L_n = F_{n-1} + F_{n+1}$ for all $n \in \mathbb{Z}_{\geq 1}$.

(c) Using Theorem 5.8.3 (Epp), derive an explicit formula for L_n.

Q2. I need to climb a flight of stairs of n steps. How many ways can I climb it if at every move I can take 1 or 2 steps?

Hint: Let s_n be the number of ways to climb n steps. Find a recurrence relation for s_n, with suitable initial values, then solve it.

Q3. Fermat's Little Theorem[1] (Theorem 8.4.10 (Epp)) states that:

(1^{st} version:) For any prime p and any integer a, $a^p \equiv a \pmod{p}$.

Alternatively:

(2^{nd} version:) For any prime p and any integer a, if $p \nmid a$ then $a^{p-1} \equiv 1 \pmod{p}$.

Verify the theorem for $p = 7, a = 4$ as well as $p = 3, a = 6$.

[1] Not to be confused with Fermat's Last Theorem, which confounded mathematicians for over 300 years until it was solved by Andrew Wiles in 1994.

Q4. In this question you will prove the mathematics behind the **RSA algorithm**, which is a widely-used public-key encryption algorithm, eg. in https web sites. It is called "public-key" because each individual publishes a public-key that anyone can use to encrypt a message, after which only that individual can decrypt the message using the corresponding private-key that is never revealed to anyone else.

The purpose of encryption is to "scramble" your message so that someone seeing it will not be able to understand it, thereby protecting its secrecy. For example, your original message, "MEET ME AT 11AM", is encrypted to become "XJ87!340XMHTLBKT", which is clearly gibberish to anyone spying on it.

The original message is called *plaintext*, and the encrypted version *ciphertext*. Decryption undoes the effect of encryption, allowing you to get back plaintext from ciphertext. Encryption should always be used when you send sensitive information over the network, eg. your credit card number or password. Your web browser should be doing this automatically.

Notation

Let $[a..b] = \{x : x \in \mathbb{Z} \wedge a \leq x \leq b\}$ [the integers in the range from a to b inclusive].

Let $a\%b = a \ mod \ b$ [the remainder when a is divided by b].

RSA Key Generation

Choose $m = pq$ where p, q are large distinct primes.

Choose $e \in \mathbb{N}$ such that $\gcd(e, (p-1)(q-1)) = 1$.

Obtain $d \in \mathbb{N}$ such that $ed \equiv 1 \pmod{(p-1)(q-1)}$ [by Bézout's Identity].

Make (m, e) public and keep (p, q, d) private.

RSA Encryption

To send a message $x \in [0..m-1]$ to someone whose public key is (m, e), encrypt it using $x^e\%m$.

RSA Decryption

On receiving a message y encrypted using your public key (m, e), decrypt it using $y^d\%m$.

Note that both encryption and decryption can be efficiently done using **repeated-squaring**. For example:

$7^{13}\%m$

$= ((7^6\%m)^2 \times 7)\%m$

$= (((7^3\%m)^2)^2 \times 7)\%m$

$= (((((7^2\%m) \times 7)\%m)^2)^2 \times 7)\%m.$

As can be seen, the number of multiplications needed is less than twice the number of bits in the exponent, since the exponent is halved at each step.

What you will prove in this question is that the decryption always works, namely that $(x^e\%m)^d\%m = x$ where m, e, d are as specified in the Key Generation phase and x is the original message as specified in the Encryption phase.

Proof sketch:

1. Let $k \in \mathbb{Z}$ such that $ed = k(p-1)(q-1)+1$. Verify that $k \geq 0$.
2. Then prove that $(x^e \% m)^d \% m = (x^{k(p-1)(q-1)} \times x) \% m$.
3. Now prove that $x^{k(p-1)(q-1)} \times x \equiv x \pmod{p}$ by splitting into two cases:
   Case 1: $p \mid x$.
   Case 2: $p \nmid x$. Hint: Use Fermat's Little Theorem here.
4. Thus, $p \mid x^{k(p-1)(q-1)} \times x - x$.
5. Thus by symmetry $q \mid x^{k(p-1)(q-1)} \times x - x$.
6. Thus $m = pq \mid x^{k(p-1)(q-1)} \times x - x$ [Why?].
7. Thus $(x^{k(p-1)(q-1)} \times x) \% m = x$ [Why?].

(Notice that the sender should ensure that x is not any of $0, 1, m-1$ otherwise it is trivial to decrypt.) Write out a formal proof of the above sketch.

Q5. Let's now use RSA to encrypt and decrypt some messages. Encode each (uppercase only) letter of the alphabet plus numerals using the following **Encoding Scheme**:

$$\text{“A”} = 2,\ \text{“B”} = 3,\ \text{“C”} = 4, \ldots,\ \text{“Z”} = 27,\ \text{“ ”} = 28,$$
$$\text{“0”} = 29,\ \text{“1”} = 30,\ \text{“2”} = 31, \ldots\ \text{“9”} = 38$$

Note that " " denotes a blank space. That is, the letter "A" is encoded as 2, the numeral "1" is encoded as 30, and so on.

Now, setup the RSA parameters with: $p = 11, q = 5, e = 3, d = 27$. These numbers are kept small so that you can use your calculator. In practice, p, q are very large primes, each one being several hundred digits long!

(a) Encrypt the plaintext message "THANK U" as follows:
   i. Encode each character, including space, in the message as a number using the Encoding Scheme above.
   ii. For each number, x, encrypt it by computing $x^e \ mod\ pq$.
   iii. You should now have a list of numbers (ciphertext). What is it?

(b) Decrypt your list of numbers in part (a) as follows:
   i. For each number, y, compute $z = y^d \ mod\ pq$. Since d is large, you should use the repeated-squaring trick mentioned in Q4, so as to keep the numbers small.
   ii. Decode each z by looking up the Encoding Scheme above.
   iii. Verify that you obtained the correct plaintext.

(c) Decrypt this ciphertext: $9, 25, 50, 36, 43, 50, 7, 10, 25, 7, 13, 33, 20$. What is the message?

Q6. A sequence a_n is defined by the third-order recurrence relation:

$$a_n = 5a_{n-1} - 8a_{n-2} + 4a_{n-3}, \text{ for } n \in \mathbb{Z}_{\geq 3}.$$

with initial values: $a_0 = a_1 = 1$ and $a_2 = 3$.

(a) (1 mark) Explicitly calculate a_3, a_4, a_5 and a_6.

(b) (4 marks) Using Mathematical Induction, prove that:

$$\sum_{r=0}^{n-1} r2^r \;=\; 2^n(n-2)+2, \text{ for all } n \in \mathbb{Z}^+.$$

(c) (5 marks) Derive an explicit closed-form formula for a_n, for all $n \in \mathbb{N}$.

Hint: Define a new sequence b_n in terms of a_n, and get a second-order linear homogeneous recurrence relation with constant coefficients for b_n. Solve it, then solve for a_n. Do not use the formula for a 3rd-order recurrence relation.

Q7. Least common multiple.

(a) Calculate $\text{lcm}(330, 156)$ by hand using prime factorization.

(b) Prove that for all positive integers a and b, $\gcd(a, b) \times \text{lcm}(a, b) = ab$. Verify this for $a = 330, b = 156$.

(c) Using the above, write code to calculate the lcm of two integers. Run your code to find $\text{lcm}(1440, 312)$.