

CAS CS 538. Problem Set 1

Due in class Wednesday, September 10, 2008, *before* the start of lecture

Reading: Read the course syllabus.

Recall the following fact from integer arithmetic: for every pair of integers a, b , there exists a unique pair of integers q, r such that $a = qb + r$ and $0 \leq r < b$. Simply put, this says that we can divide two integers and get a unique quotient q and remainder r . If $r = 0$, then $a = qb$, and we say that “ b divides a ” and write $b|a$.

Define $a \bmod b$ (pronounced “ a modulo b ”) to be r . We say $a_1 \equiv a_2 \pmod{b}$ (pronounced “ a_1 is congruent to a_2 modulo b ”) if and only if $a_1 \bmod b = a_2 \bmod b$.

Note the somewhat confusing notation: $\bmod b$ by itself is a binary operator that gives you a remainder (like “%” in C), while \pmod{b} in parentheses is not an operator, but an “explainer” for the binary relation \equiv . Note also that it is common to omit \pmod{b} when it is clear from the context what \equiv means.

Problem 1. (25 points) You may use the above fact about unique division with remainder, elementary axioms (such as commutativity, distributivity, etc.) and definitions of \bmod and \equiv in your proofs below. You may also use for each statement the statements that appear before it, whether or not you have succeeded in proving them (they’ll be graded independently). Do not use anything else; if you feel you need to use some other fact that you cannot prove, clearly state the fact and explain that you don’t know how to prove it. Please use words like “because” and “therefore” in your proofs: a collection of statements without any logical connectors is impossible to understand.

The congruences in parts (a)–(d) are modulo b and “ \pmod{b} ” is omitted.

- (a) Prove that $a_1 \equiv a_2$ if and only if $b|(a_1 - a_2)$.
- (b) Prove that $a \bmod b \equiv a$.
- (c) Prove that $(a_1 \bmod b) + (a_2 \bmod b) \equiv a_1 + a_2$ and $(a_1 \bmod b)(a_2 \bmod b) \equiv a_1 a_2$. This should convince you that when performing modular addition and multiplication, it doesn’t matter whether or not you first modular reduce the operands.
- (d) Prove that $-a \equiv b - a$. Thus, to subtract a is the same as to add $b - a$.
- (e) Using the above, compute $249^{16} \bmod 251$ without using numbers longer than three decimal digits. Do not use a calculator. Show work. (*Hint:* $249 \equiv -2 \pmod{251}$.)

Problem 2. (15 points)

- (a) Compute $7^{64} \bmod 23$ (show work; do not use a calculator).
(Hint: compute $7^2, 7^4, 7^8, \dots \bmod 23$ first.)
- (b) Compute $7^{71} \bmod 23$ (show work; do not use a calculator).
(Hint: use some of the intermediate results from the previous part.)
- (c) Describe an efficient algorithm to compute $a^b \bmod c$ for positive integers a, b, c .

Problem 3. (20 points) Let $p > 2$ be a prime (the definition of “prime” here is the following: if $p|xy$ then $p|x$ or $p|y$). All congruences in this problem are modulo p . Prove that:

(a) For any integers $a \not\equiv 0$, r and s , if $ra \equiv sa$, then $r \equiv s$. (Note that you don’t yet know that “division” modulo p exists.)

(b) For any integer $a \not\equiv 0$, the values $a \bmod p, 2a \bmod p, 3a \bmod p, \dots, (p-1)a \bmod p$ hit every element in the set $1, 2, 3, \dots, p-1$ exactly once. (Hint: use the previous part.)

(c) Fermat’s little theorem: if $a \not\equiv 0 \pmod{p}$, then $a^{p-1} \equiv 1 \pmod{p}$. (Hint: multiply together the $p-1$ values $a, 2a, 3a, \dots, (p-1)a$ and use the previous part.)

(d) Pronounce Fermat correctly (it’s like “fur+Ma” and **not** like “fur+Matt”).

(e) Show that every non-zero element modulo p has an inverse: for every $a \not\equiv 0$, there exists b such that $ab \equiv 1$. Thus, division modulo p makes sense.

Problem 4. (20 points) The Vernam cipher (one-time-pad) works for any message space of size 2^k , and has key space of size 2^k as well. This works fine for sizes that are powers of 2.

For any given message space m_1, m_2, \dots, m_s of size s (not necessarily a power of two), design a cryptosystem that has key space of size exactly s . Prove that it is perfectly secure.

Problem 5. (20 points) Bob is establishing an account with Alice, a discount on-line broker. He wants his trading to be private. Since both Alice and Bob have heard that the one-time pad (a.k.a. the Vernam cipher) is a very secure cryptosystem they generate a 96-bit-long random pad k for Bob to use in the future for encrypting all his future buy/sell orders to Alice.

They agree on the following format for each order: first, Bob writes down a single character, either ‘B’ for ‘Buy’ or ‘S’ for ‘Sell.’ Then he puts a single space, followed by a five-digit decimal number for the number of shares he wants to buy or sell (if he doesn’t need to use all five digits, he puts zeros in the front). Finally, he puts another space followed by the four-letter ticker symbol of the stock he wants to buy or sell (if not all four letters are needed, he puts spaces in the front). Thus, for example, ‘ ‘B.00100_MSFT’ ’ means “Buy 100 shares of Microsoft” and ‘ ‘S.25000_GOGL’ ’ means “Sell 25,000 shares of Google.”

(a) Are the orders securely encrypted — that is, can an eavesdropper Eve obtain any information about what’s being traded from observing the messages from Bob to Alice? Why or why not?

(b) Suppose now an eavesdropper Eve is replaced by an attacker Moe who can not only read the ciphertexts, but also modify them in transit. However, Bob sends only one order to Alice using k . If it is a buy order, can Moe change it to a sell order? Can Moe change the order if he doesn’t know whether it’s buy or sell? Why or why not?