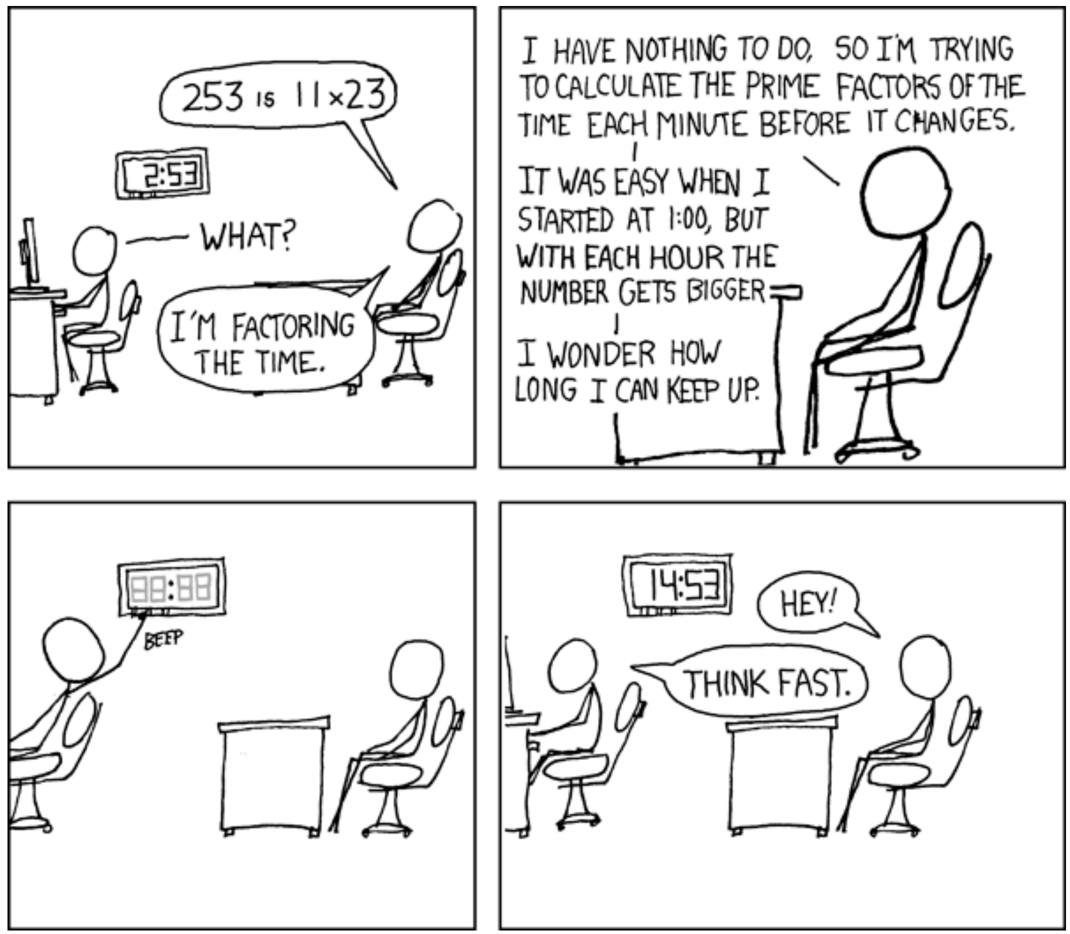
CSE 311: Foundations of Computing

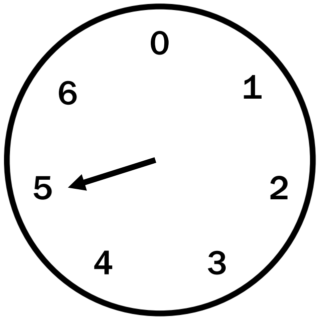
Lecture 12: Primes, GCD



Last Time: Modular Arithmetic

$$(a + b) \bmod 7$$

$$(a \times b) \bmod 7$$



+	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

Replace number line with a clock. Taking *m* steps returns back to the same place.

Form of arithmetic using only a finite set of numbers {0, 1, 2, 3, ..., *m* – 1}

Unclear (so far) that modular arithmetic has the same properties as ordinary arithmetic....

Last Time: Modular Arithmetic

Idea: Find replacement for "=" that works for modular arithmetic

"=" on ordinary numbers allows us to solve problems, e.g.

- add / subtract numbers from both sides of equations
- substitute "=" values in equations

Definition: "a is congruent to b modulo m"

For $a, b, m \in \mathbb{Z}$ with $m > 0$

$$a \equiv b \pmod{m} \leftrightarrow m \mid (a - b)$$

Equivalently, $a \equiv b \pmod{m}$ iff $a = b + km$ for some $k \in \mathbb{Z}$.

Last Time: Modular Arithmetic

Definition: "a is congruent to b modulo m"

For $a, b, m \in \mathbb{Z}$ with $m > 0$

$$a \equiv b \pmod{m} \leftrightarrow m \mid (a - b)$$

Equivalently, $a \equiv b \pmod{m}$ **iff** $a = b + km$ **for some** $k \in \mathbb{Z}$**.**

$\boldsymbol{a \equiv b \pmod{m}}$ if and only if $\boldsymbol{a \bmod m = b \bmod m}$.

I.e., $\boldsymbol{a}$ **and** $\boldsymbol{b}$ **are congruent modulo m iff** $\boldsymbol{a}$ **and** $\boldsymbol{b}$ **steps go to the same spot on the "clock" with m numbers**

Last Time: Modular Arithmetic: Properties

If $\boldsymbol{a} \equiv \boldsymbol{b} \pmod{\boldsymbol{m}}$ and $\boldsymbol{b} \equiv \boldsymbol{c} \pmod{\boldsymbol{m}}$, then $\boldsymbol{a} \equiv \boldsymbol{c} \pmod{\boldsymbol{m}}$

If $\boldsymbol{a} \equiv \boldsymbol{b} \pmod{\boldsymbol{m}}$ and $\boldsymbol{c} \equiv \boldsymbol{d} \pmod{\boldsymbol{m}}$, then $\boldsymbol{a}+\boldsymbol{c} \equiv \boldsymbol{b}+\boldsymbol{d} \pmod{\boldsymbol{m}}$

Corollary: If $\boldsymbol{a} \equiv \boldsymbol{b} \pmod{\boldsymbol{m}}$, then $\boldsymbol{a}+\boldsymbol{c} \equiv \boldsymbol{b}+\boldsymbol{c} \pmod{\boldsymbol{m}}$

If $\boldsymbol{a} \equiv \boldsymbol{b} \pmod{\boldsymbol{m}}$ and $\boldsymbol{c} \equiv \boldsymbol{d} \pmod{\boldsymbol{m}}$, then $\boldsymbol{ac} \equiv \boldsymbol{bd} \pmod{\boldsymbol{m}}$

Corollary: If $\boldsymbol{a} \equiv \boldsymbol{b} \pmod{\boldsymbol{m}}$, then $\boldsymbol{ac} \equiv \boldsymbol{bc} \pmod{\boldsymbol{m}}$

Last Time: Modular Arithmetic: Properties

If $\boldsymbol{a} \equiv \boldsymbol{b} \ (\mathbf{mod}\ \boldsymbol{m})$ and $\boldsymbol{b} \equiv \boldsymbol{c} \ (\mathbf{mod}\ \boldsymbol{m})$, then $\boldsymbol{a} \equiv \boldsymbol{c} \ (\mathbf{mod}\ \boldsymbol{m})$

If $\boldsymbol{a} \equiv \boldsymbol{b} \ (\mathbf{mod}\ \boldsymbol{m})$, then $\boldsymbol{a} + \boldsymbol{c} \equiv \boldsymbol{b} + \boldsymbol{c} \ (\mathbf{mod}\ \boldsymbol{m})$

If $\boldsymbol{a} \equiv \boldsymbol{b} \ (\mathbf{mod}\ \boldsymbol{m})$, then $\boldsymbol{ac} \equiv \boldsymbol{bc} \ (\mathbf{mod}\ \boldsymbol{m})$

"≡" allows us to solve problems in modular arithmetic, e.g.

- add / subtract numbers from both sides of equations
- chains of "≡" values shows first and last are "≡"
- substitute "≡ " values in equations (not *fully* proven yet)

Basic Applications of mod

- Two's Complement
- Hashing
- Pseudo random number generation

n-bit Unsigned Integer Representation

- Represent integer x as sum of powers of 2:

  If $\sum_{i=0}^{n-1} b_i 2^i$ where each $b_i \in \{0,1\}$

  then representation is $b_{n\text{-}1}...b_2\, b_1\, b_0$

  99 = 64 + 32 + 2 + 1

  18 = 16 + 2

- For n = 8:

  99: 0110 0011

  18: 0001 0010

Easy to implement arithmetic $\mathbf{mod\ 2}^n$
... just throw away bits n+1 and up

Sign-Magnitude Integer Representation

n-bit signed integers

Suppose that $-2^{n-1} < x < 2^{n-1}$

First bit as the sign, $n - 1$ bits for the value

99 = 64 + 32 + 2 + 1

18 = 16 + 2

For n = 8:

99: 0110 0011

-18: 1001 0010

Any problems with this representation?

Two's Complement Representation

n bit signed integers, first bit will still be the sign bit

Suppose that $0 \leq x < 2^{n-1}$,

x is represented by the binary representation of x

Suppose that $0 \leq x \leq 2^{n-1}$,

$-x$ is represented by the binary representation of $-x + 2^n$

Key property: Twos complement representation of any number $\boldsymbol{y}$ is equivalent to $\boldsymbol{y}$ $\mathbf{mod}\ \mathbf{2^n}$ so arithmetic works $\mathbf{mod}\ \mathbf{2^n}$

99 = 64 + 32 + 2 + 1
18 = 16 + 2

For n = 8:
99: 0110 0011
-18: 1110 1110

Sign-Magnitude vs. Two's Complement

-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
1111	1110	1101	1100	1011	1010	1001	0000	0001	0010	0011	0100	0101	0110	0111

Sign-bit

-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
1000	1001	1010	1011	1100	1101	1110	1111	0000	0001	0010	0011	0100	0101	0110	0111

Two's complement

Two's Complement Representation

- For $0 < x \leq 2^{n-1}$, $-x$ is represented by the binary representation of $2^n - x$
  - That is, the two's complement representation of any number y has the same value as y modulo 2^n.

Two's Complement Representation

- For $0 < x \leq 2^{n-1}$, $-x$ is represented by the binary representation of $2^n - x$
  - That is, the two's complement representation of any number y has the same value as y modulo 2^n.

- To compute this: Flip the bits of x then add 1:
  - All 1's string is $2^n - 1$, so

    Flip the bits of x $\equiv$ replace x by $2^n - 1 - x$

    Then add 1 to get $2^n - x$

Hashing

Scenario:

Map a small number of data values from a large domain $\{0, 1, \ldots, M-1\}$ ...

...into a small set of locations $\{0,1, \ldots, n-1\}$ so one can quickly check if some value is present

- $\text{hash}(x) = x \bmod p$ for p a prime close to n
  - or $\text{hash}(x) = (ax + b) \bmod p$
- Depends on all of the bits of the data
  - helps avoid collisions due to similar values
  - need to manage them if they occur

Pseudo-Random Number Generation

Linear Congruential method

$$x_{n+1} = (a\, x_n + c) \bmod m$$

Choose random x_0, a, c, m and produce a long sequence of x_n's

More Number Theory

Primes and GCD

Primality

An integer *p* greater than 1 is called *prime* if the only positive factors of *p* are 1 and *p*.

A positive integer that is greater than 1 and is not prime is called *composite*.

Fundamental Theorem of Arithmetic

Every positive integer greater than 1 has a "unique" prime factorization

48 = 2 • 2 • 2 • 2 • 3
591 = 3 • 197
45,523 = 45,523
321,950 = 2 • 5 • 5 • 47 • 137
1,234,567,890 = 2 • 3 • 3 • 5 • 3,607 • 3,803

Euclid's Theorem

There are an infinite number of primes.

Proof by contradiction:

Suppose that there are only a finite number of primes and call the full list $p_1, p_2, \ldots, p_n$.

Euclid's Theorem

There are an infinite number of primes.

Proof by contradiction:

Suppose that there are only a finite number of primes and call the full list $p_1, p_2, \ldots, p_n$.

Define the number $P = p_1 \cdot p_2 \cdot p_3 \cdot \cdots \cdot p_n$ and let $Q = P + 1$.

Euclid's Theorem

There are an infinite number of primes.

Proof by contradiction:

Suppose that there are only a finite number of primes and call the full list $p_1, p_2, \ldots, p_n$.

Define the number $P = p_1 \cdot p_2 \cdot p_3 \cdot \cdots \cdot p_n$ and let $Q = P + 1$. (Note that $Q > 1$.)

Case 1: Q is prime: Then Q is a prime different from all of $p_1, p_2, \ldots, p_n$ since it is bigger than all of them.

Case 2: Q is not prime: Then Q has some prime factor p (which must be in the list). Therefore $p|P$ and $p|Q$ so $p|(Q - P)$ which means that $p|1$.

Both cases are contradictions,
so the assumption is false (proof by cases). ∎

Famous Algorithmic Problems

- Primality Testing
  - Given an integer n, determine if n is prime
- Factoring
  - Given an integer n, determine the prime factorization of n

Factoring

Factor the following 232 digit number [RSA768]:

1230186684530117755130494958384962720772
8535695953347921973224521517264005072636
5751874520219978646938995647494277406384
5925192557326303453731548268507917026122
1429134616704292143116022212404792747377
9408066535141959745985690214 3413

1230186684530117755130494958384962720772853569595334
7921973224521517264005072636575187452021997864693899564749427740638459251925573263034537315482685079170261221429134616704292143116022212404792747377940806653514195974598569021434 13

=

33478071698956898786044169848212690817704794983713768568912431388982883793878002287614711652531743087737814467999489

×

36746043666799590428244633799627952632279158164343087642676032283815739666511279233373417143396810270092798736308917

Greatest Common Divisor

GCD(a, b):

Largest integer d such that $d \mid a$ and $d \mid b$

- GCD(100, 125) =
- GCD(17, 49) =
- GCD(11, 66) =
- GCD(13, 0) =
- GCD(180, 252) =

GCD and Factoring

$a = 2^3 \cdot 3 \cdot 5^2 \cdot 7 \cdot 11 = 46{,}200$

$b = 2 \cdot 3^2 \cdot 5^3 \cdot 7 \cdot 13 = 204{,}750$

$$GCD(a, b) = 2^{\min(3,1)} \cdot 3^{\min(1,2)} \cdot 5^{\min(2,3)} \cdot 7^{\min(1,1)} \cdot 11^{\min(1,0)} \cdot 13^{\min(0,1)}$$

Factoring is expensive!

Can we compute GCD(a,b) without factoring?

Useful GCD Fact

If *a* and *b* are positive integers, then

$$\gcd(a,b) = \gcd(b, a \bmod b)$$

Proof:

By definition of mod, $a = qb + (a \bmod b)$ for some integer $q = a \operatorname{div} b$.

Suppose $d|a$ and $d|b$.
Then $a = kd$ and $b = jd$ for some integers k and j.

Therefore $(a \bmod b) = a - qb = kd - qjd = (k - qj)d$.
So, $d|(a \bmod b)$, and since $d|b$ we must have $d \mid \gcd(b, a \bmod b)$.

Suppose $e|b$ and $e \mid (a \bmod b)$.
Then $b = me$ and $(a \bmod b) = ne$ for some integers m and n.
Therefore $a = qb + (a \bmod b) = qme + ne = (qm + n)e$. So $e|a$.

Since they have the same common divisors, $\gcd(a, b) = \gcd(b, a \bmod b)$. ■

Another simple GCD fact

If a is a positive integer, gcd($a,0$) = a.