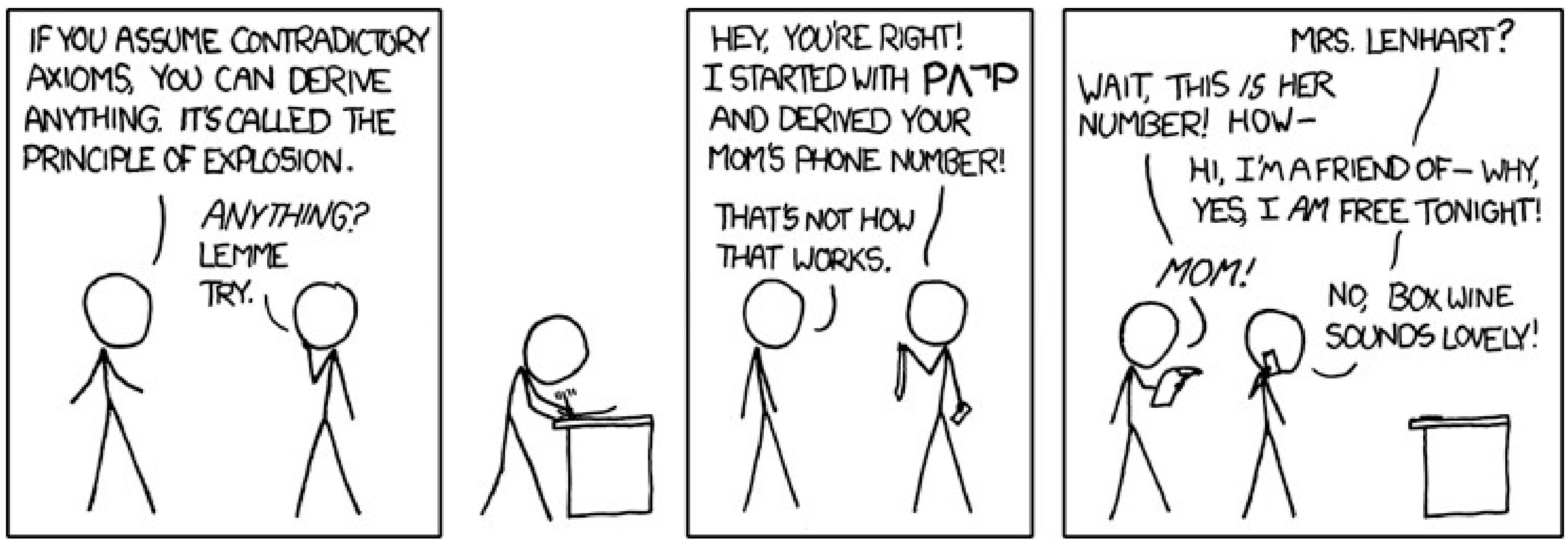

Number Theory

CSE 311
Lecture 7

What we have proven so far:

- Let a, b, c, d and $m > 0$ be integers.

- If $a \equiv_m b$, then $b \equiv_m a$.
- If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.
- If $a \equiv_m b$ and $c \equiv_m d$, then $ac \equiv_m bd$.
- If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.

Todo:

- $a \equiv_m b$ if and only if $a \% m = b \% m$.

Claim 5:

Claim 5: For integers a, b and $m > 0$, $a \equiv_m b$ if and only if $a \,\%\, m = b \,\%\, m$.

For integers a, b and $m > 0$, $a \equiv_m b$ if and only if $a \,\%\, m = b \,\%\, m$.

$\Rightarrow$ Let $a, b, m > 0$ be arbitrary integers, and suppose $a \equiv_m b$. Then $m \mid (a - b)$. So there exists some integer k such that $a - b = km$. So $a = km + b$.

By the Division Theorem, $a = qm + (a \,\%\, m)$ for some integer q, where $0 \leq a \,\%\, m < m$. Thus:

$$km + b = qm + (a \,\%\, m)$$

$$b = qm - km + (a \,\%\, m)$$

$$b = (q - k)m + (a \,\%\, m)$$

By the Division Theorem again, we have that $b \,\%\, m = a \,\%\, m$.

Since a, b, m were arbitrary, the claim holds.

> For integers a, b and $m > 0$, $a \equiv_m b$ if and only if $a \,\%\, m = b \,\%\, m$.

$\Leftarrow$ Let $a, b, m > 0$ be arbitrary integers, and suppose $a \,\%\, m = b \,\%\, m$. By the Division Theorem, $a = mq + (a \,\%\, m)$ for some integer q, and $b = ms + (b \,\%\, m)$ for some integer s. Thus:

$$a - b = \big(mq + (a \,\%\, m)\big) - \big(ms + (b \,\%\, m)\big)$$

$$a - b = mq - ms + (a \,\%\, m) - (b \,\%\, m)$$

$$a - b = m(q - s)$$

Since q, s are integers, $q - s$ is an integer. So $m \mid (a - b)$. So $a \equiv_m b$.

Since a, b, m were arbitrary, the claim holds.

Summary: Properties of Mod

- Let a, b, c, d and $m > 0$ be integers.

- If $a \equiv_m b$, then $b \equiv_m a$.
- If $a \equiv_m b$ and $c \equiv_m d$, then $a + c \equiv_m b + d$.
- If $a \equiv_m b$ and $c \equiv_m d$, then $ac \equiv_m bd$.
- If $a \equiv_m b$ and $b \equiv_m c$, then $a \equiv_m c$.
- $a \equiv_m b$ if and only if $a \,\%\, m = b \,\%\, m$.

Another contrapositive example

Another Proof

For all integers, a, b, c: Show that if $a \nmid (bc)$ then $a \nmid b$ or $a \nmid c$.

Proof:

Let a, b, c be arbitrary integers, and suppose $a \nmid (bc)$.

Then there is not an integer z such that $az = bc$

…

So $a \nmid b$ or $a \nmid c$

Another Proof

For all integers, a, b, c**:** Show that if $a \nmid (bc)$ then $a \nmid b$ or $a \nmid c$.

Proof:

Let a, b, c be arbitrary integers, and suppose $a \nmid (bc)$**.**

Then there is not an integer z such that $az = bc$

…

So $a \nmid b$ or $a \nmid c$

Another Proof

For all integers, a, b, c: Show that if $a \nmid (bc)$ then $a \nmid b$ or $a \nmid c$.

There has to be a better way!

If only there were some equivalent implication…

One where we could negate everything…

Take the contrapositive of the statement:

For all integers, a, b, c: Show if $a|b$ and $a|c$ then $a|(bc)$.

By contrapositive

Claim: For all integers, a, b, c: Show that if $a \nmid (bc)$ then $a \nmid b$ or $a \nmid c$.

We argue by contrapositive.

Let a, b, c be arbitrary integers, and suppose $a|b$ and $a|c$.

Therefore $a|bc$

By contrapositive

Claim: For all integers, a, b, c: Show that if $a \nmid (bc)$ then $a \nmid b$ or $a \nmid c$.

We argue by contrapositive.

Let a, b, c be arbitrary integers, and suppose $a|b$ and $a|c$.

By definition of divides, $ax = b$ and $ay = c$ for integers x and y.

Multiplying the two equations, we get $axay = bc$

Since a, x, y are all integers, xay is an integer. Applying the definition of divides, we have $a|bc$.

Logical Ordering

Logical Ordering

- When doing a proof, we often work from both sides…
- But we have to be careful!
- When you read from top to bottom, every step has to follow only from what's before it, not after it.

- Suppose our target is q and I know $q \rightarrow p$ and $r \rightarrow q$.
- What can I put as a "new target?"

Logical Ordering

- So why have all our prior steps been ok backward?

- They’ve all been either:
  - A definition (which is always an “if and only if”)
  - An algebra step that is an “if and only if”

- Even if your steps are “if and only if” you still have to put everything in order – start from your assumptions, and only assert something once it can be shown.

A bad proof (Backwards Proof)

Claim: if x is positive then $x + 5 = -x - 5$.

$$x + 5 = -x - 5.$$
$$x + 5 = -x - 5$$
$$|x + 5| = |-x - 5|$$
$$|x + 5| = |-(x + 5)|$$
$$|x + 5| = |x + 5|$$

$$0 = 0$$

This claim is false – if you're trying to do algebra, you need to start with an equation you know (say $x = x$ or $2 = 2$ or $0 = 0$) and expand to the equation you want.

Primes & GCD

Algorithmic Problems

- Multiplication
  - Given primes $p_1, p_2, \ldots, p_k$, calculate their product $p_1 p_2 \ldots p_k$
- Factoring
  - Given an integer n, determine the prime factorization of n

Factoring

Factor the following 232 digit number [RSA768]:

123018668453011775513049495838496272
077285356959533479219732245215172640
050726365751874520219978646938995647
494277406384592519255732630345373154
826850791702612214291346167042921431
160222124047927473779408066535141959
7459856902143413

1230186684530117755130494958384962720772853569595334792197322452151726400507263657518745202199786469389956474942774063845925192557326303453731548268507917026122142913461670429214311602221240479274737794080665351419597459856902143413

=

334780716989568987860441698482126908177047949837137685689124313889828837938780022876147116525317430877378144679994 89

×

367460436667995904282446337996279526322791581643430876426760322838157396665112792333734171433968102700927987363089 17

Famous Algorithmic Problems

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

- **Factoring** is hard
  - (on a classical computer)
- **Primality Testing** is easy

Prime and Composite

- Definition:

An integer $p > 1$ is prime iff its only positive divisors are 1 and p.

- An integer $p > 1$ is composite iff it is not prime.

Fundamental Theorem of Arithmetic

Every Positive integer greater than 1 has a “unique” prime factorization:

e.g: 42 = 2 * 2 * 2 * 2 * 3, 591 = 3 * 197, ect…

Greatest Common Divisor

- Definition:

The Greatest Common Divisor of integers a and b (denoted $\mathbf{gcd}(a, b)$) is the largest integer c such that $c \mid a$ and $c \mid b$.

- Useful Fact: Let a be a positive integer. The GCD(a, 0) = a
- For Example:

$\gcd(99,18) = 9$

$\gcd(100, 125) = 25$

$\gcd(7,11) = 1$

$\gcd(13,0) = 13$

Calculating the GCD: Approach 1

- Fundamental Theorem of Arithmetic: Every positive integer greater than 1 has a unique prime factorization.

- Approach 1 to finding $\gcd(a, b)$:

1. Find the prime factorization of a
2. Find the prime factorization of b
3. Identify all common prime factors.
4. Multiply the common prime factors together. This is the GCD.

VERY INEFFICIENT

Calculating the GCD: Approach 2

- Claim: For positive integers a, b, $\text{gcd}(a, b) = \text{gcd}(b, a \% b)$.

- For example:
- $\text{gcd}(10, 6) = \text{gcd}(6,4)$
- $\text{gcd}(110,30) = \text{gcd}(30,20)$

- We'll prove this in a minute. But first: how can we use this fact to devise an algorithm for computing $\text{gcd}(a, b)$?

Calculating the GCD: Approach 2

- Euclid’s Algorithm. To find $\mathrm{gcd}(a, b)$:
  - Repeatedly use $\mathrm{gcd}(a, b) = \mathrm{gcd}(b, a \,\%\, b)$ to reduce numbers
  - Stop once you reach $\mathrm{gcd}(g, 0)$. Return g.

- For Example:

$$\begin{aligned}\mathrm{gcd}(660{,}126) &= \mathrm{gcd}(126{,}30)\\ &= \mathrm{gcd}(30{,}6)\\ &= \mathrm{gcd}(6{,}0)\\ &= 6\end{aligned}$$

Efficient

Euclid’s Algorithm in Java

```
- // assumes a >= 0 and b >= 0
- public int gcd(int a, int b) {
-    if (b == 0) {
-          return a;
-    } else {
-          return gcd(b, a % b);
-    }
- }
```

Proof of Claim

- Claim: For positive integers a, b, $\gcd(a, b) = \gcd(b, a \% b)$.

- How do you show that two GCDs are equal?

• First consider some arbitrary common divisor of a and b, call it d. Prove that d is a divisor of $a \% b$.

• Then consider some arbitrary common divisor of b and $a \% b$, call it d. Prove that d is a divisor of a.

• Thus a and b have the same common divisors as b and $a \% b$. So their GCDs are equal.

Claim: For positive integers a, b, $\gcd(a, b) = \gcd(b, a \,\%\, b)$.

Let a, b be arbitrary positive integers. By the Division Theorem, $a = qb + (a \,\%\, b)$ for some int q.

Let d be arbitrary. Suppose $d \mid b$ and $d \mid a \,\%\, b$. We aim to show that $d \mid a$. By definition of divides, $b = kd$ and $a \,\%\, b = jd$ for some integers k, j. Then it follows that:

$$a = qb + (a \,\%\, b) = q \cdot kd + jd = d(qk + j)$$

Since q, k, j are integers, $qk + j$ is an integer. So $d \mid a$.

Now suppose $d \mid a$ and $d \mid b$. We aim to show that $d \mid a \,\%\, b$. By definition of divides, $a = md$ and $b = nd$ for some integers m, n. Then it follows that:

$$a \,\%\, b = a - qb = md - qnd = d(m - qn)$$

Since q, m, n are integers, $m - qn$ is an integer. So $d \mid a \,\%\, b$.

Thus a and b have the same common divisors as b and $a \,\%\, b$. So $\gcd(a, b) = \gcd(b, a \,\%\, b)$. Since a, b were arbitrary, the claim holds.

Bézout's theorem

If a and b are positive integers, then there exist integers $\boldsymbol{s}$ and $\boldsymbol{t}$ such that

$$\gcd(a,b) = sa + tb.$$

$$\forall a\ \forall b\ ((a > 0 \land b > 0) \rightarrow \exists s\ \exists t\ (\gcd(a,b) = sa + tb))$$

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 1 (Compute GCD & Keep Tableau Information):

a b b a mod b = r b r

$\gcd(35, 27) = \gcd(27, 35 \bmod 27) = \gcd(27, 8)$

a = q * b + r

$35 = 1 * 27 + 8$

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 1 (Compute GCD & Keep Tableau Information):

a b	b a mod b = r	b r	a = q * b + r
$\gcd(35, 27)$	$= \gcd(27, 35 \bmod 27)$	$= \gcd(27, 8)$	$35 = 1 * 27 + 8$
	$= \gcd(8, 27 \bmod 8)$	$= \gcd(8, 3)$	$27 = 3 * 8 + 3$
	$= \gcd(3, 8 \bmod 3)$	$= \gcd(3, 2)$	$8 = 2 * 3 + 2$
	$= \gcd(2, 3 \bmod 2)$	$= \gcd(2, 1)$	$3 = 1 * 2 + 1$
	$= \gcd(1, 2 \bmod 1)$	$= \gcd(1, 0)$	

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 2 (Solve the equations for r):

a = q * b + r

$$35 = 1 * 27 + 8$$
$$27 = 3 * 8 \quad + 3$$
$$8 \quad = 2 * 3 \quad + 2$$
$$3 \quad = 1 * 2 \quad + 1$$

r = a -- q * b

$$8 = 35 - 1 * 27$$

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 2 (Solve the equations for r):

a = q * b + r

$35 = 1 * 27 + 8$
$27 = 3 * 8 + 3$
$8 = 2 * 3 + 2$
$3 = 1 * 2 + 1$

r = a -- q * b

$8 = 35 - 1 * 27$
$3 = 27 - 3 * 8$
$2 = 8 - 2 * 3$
$1 = 3 - 1 * 2$

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 3 (Backward Substitute Equations):

$8 = 35 - 1 * 27$

$3 = 27 - 3 * 8$

$2 = 8 - 2 * 3$

$1 = 3 - 1 * 2$

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 3 (Backward Substitute Equations):

$8 = 35 - 1 * 27$

$3 = 27 - 3 * 8$

$2 = 8 - 2 * 3$

$1 = 3 - 1 * 2$

Plug in the def of 2

$$1 = 3 - 1 * (8 - 2 * 3)$$
$$= 3 - 8 + 2 * 3$$
$$= (-1) * 8 + 3 * 3$$

Re-arrange into 3's and 8's

Extended Euclidean algorithm

- Can use Euclid's Algorithm to find s, t such that

$$\gcd(a, b) = sa + tb$$

Step 3 (Backward Substitute Equations):

$8 = 35 - 1 * 27$

$3 = 27 - 3 * 8$

$2 = 8 - 2 * 3$

$1 = 3 - 1 * 2$

Plug in the def of 2

$$1 = 3 - 1 * (8 - 2 * 3)$$
$$= 3 - 8 + 2 * 3$$
$$= (-1) * 8 + 3 * 3$$

Re-arrange into 3's and 8's

Plug in the def of 3

$$= (-1) * 8 + 3 * (27 - 3 * 8)$$
$$= (-1) * 8 + 3 * 27 + (-9) * 8$$
$$= 3 * 27 + (-10) * 8$$

Re-arrange into 8's and 27's

Extended Euclidean algorithm

- **Can use Euclid's Algorithm to find s, t such that**

$$\gcd(a, b) = sa + tb$$

Step 3 (Backward Substitute Equations):

$8 = 35 - 1 * 27$

$3 = 27 - 3 * 8$

$2 = 8 - 2 * 3$

$1 = 3 - 1 * 2$

Plug in the def of 2

$$1 = 3 - 1 * (8 - 2 * 3)$$
$$= 3 - 8 + 2 * 3$$
$$= (-1) * 8 + 3 * 3$$

Re-arrange into 3's and 8's

Plug in the def of 3

$$= (-1) * 8 + 3 * (27 - 3 * 8)$$
$$= (-1) * 8 + 3 * 27 + (-9) * 8$$
$$= 3 * 27 + (-10) * 8$$

Re-arrange into 8's and 27's

$$= 3 * 27 + (-10) * (35 - 1 * 27)$$
$$= 3 * 27 + (-10) * 35 + 10 * 27$$
$$= 13 * 27 + (-10) * 35$$

Re-arrange into 27's and 35's

Multiplicative inverse $\bmod m$

Let $0 \leq a, b < m$. Then, b is the *multiplicative inverse of* a *(modulo* m*)* iff $ab \equiv_m 1$.

x	0	**1**	**2**	**3**	**4**	**5**	**6**
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

mod 7

x	0	**1**	2	**3**	4	5	6	**7**	8	**9**
0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9
2	0	2	4	6	8	0	2	4	6	8
3	0	3	6	9	2	5	8	1	4	7
4	0	4	8	2	6	0	4	8	2	6
5	0	5	0	5	0	5	0	5	0	5
6	0	6	2	8	4	0	6	2	8	4
7	0	7	4	1	8	5	2	9	6	3
8	0	8	6	4	2	0	8	6	4	2
9	0	9	8	7	6	5	4	3	2	1

mod 10

Multiplicative inverse $\mathrm{mod}\ m$

Suppose $\gcd(a, m) = 1$

By Bézout's Theorem, there exist integers s and t

such that $sa + tm = 1$.

s is the multiplicative inverse of a (modulo m):

$$1 = sa + tm \equiv_m sa$$

So... we can compute multiplicative inverses with the extended Euclidean algorithm

These inverses let us solve modular equations...

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Find multiplicative inverse of 7 modulo 26

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Find multiplicative inverse of 7 modulo 26

$$\gcd(26, 7) = \gcd(7, 5) = \gcd(5, 2) = \gcd(2, 1) = 1$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

$$\gcd(26,7) = \gcd(7,5) = \gcd(5,2) = \gcd(2,1) = 1$$

$$26 = 3 * 7 + 5$$
$$7 = 1 * 5 + 2$$
$$5 = 2 * 2 + 1$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ **Find multiplicative inverse of 7 modulo 26**

$$\gcd(26,7) = \gcd(7,5) = \gcd(5,2) = \gcd(2,1) = 1$$

$$26 = 3 * 7 + 5 \qquad 5 = 26 - 3 * 7$$

$$7 = 1 * 5 + 2 \qquad 2 = 7 - 1 * 5$$

$$5 = 2 * 2 + 1 \qquad 1 = 5 - 2 * 2$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Find multiplicative inverse of 7 modulo 26

$$\gcd(26,7) = \gcd(7,5) = \gcd(5,2) = \gcd(2,1) = 1$$

$$26 = 3 * 7 + 5 \qquad 5 = 26 - 3 * 7$$
$$7 = 1 * 5 + 2 \qquad 2 = 7 - 1 * 5$$
$$5 = 2 * 2 + 1 \qquad 1 = 5 - 2 * 2$$

$$\begin{aligned} 1 &= 5 - 2 * (7 - 1 * 5) \\ &= (-2) * 7 + 3 * 5 \\ &= (-2) * 7 + 3 * (26 - 3 * 7) \\ &= (-11) * 7 + 3 * 26 \end{aligned}$$

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Find multiplicative inverse of 7 modulo 26

$$\gcd(26, 7) = \gcd(7, 5) = \gcd(5, 2) = \gcd(2, 1) = 1$$

$$26 = 3 * 7 + 5 \qquad 5 = 26 - 3 * 7$$

$$7 = 1 * 5 + 2 \qquad 2 = 7 - 1 * 5$$

$$5 = 2 * 2 + 1 \qquad 1 = 5 - 2 * 2$$

$$\begin{aligned} 1 &= 5 - 2 * (7 - 1 * 5) \\ &= (-2) * 7 + 3 * 5 \\ &= (-2) * 7 + 3 * (26 - 3 * 7) \\ &= (-11) * 7 + 3 * 26 \end{aligned}$$

Now $(-11) \bmod 26 = 15.$ ← "<u>the</u>" multiplicative inverse

(-11 is also "a" multiplicative inverse)

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Find multiplicative inverse of 7 modulo 26**… it's** 15**.**

Multiplying both sides by 15 **gives**

$$15 \cdot 7\mathrm{x} \equiv_{26} 15 \cdot 3$$

Simplify on both sides to get

$$\mathrm{x} \equiv_{26} 15 \cdot 7\mathrm{x} \equiv_{26} 15 \cdot 3 \equiv_{26} 19$$

So, <u>all</u> solutions of this congruence are numbers of the form $x = 19 + 26k$ **for some** $k \in \mathbb{Z}$**.**

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$

Conversely, suppose that $x \equiv_{26} 19$.

Multiplying both sides by 7 **gives**

$$7\text{x} \equiv_{26} 7 \cdot 19$$

Simplify on right to get

$$7\text{x} \equiv_{26} 7 \cdot 19 \equiv_{26} 3$$

So, <u>all</u> numbers of form $x = 19 + 26k$ **for any** $k \in \mathbb{Z}$ **are solutions of this equation.**

Example: Solve a Modular Equation

Solve: $7x \equiv_{26} 3$ (on HW or exams)

Step 1. Find multiplicative inverse of 7 modulo 26

$$1 \;=\; \dots \;=\; (-11) * 7 \;+\; 3 * 26$$

Since $(-11) \bmod 26 = 15$, the inverse of 7 is 15.

Step 2. Multiply both sides and simplify

Multiplying by 15, we get $x \equiv_{26} 15 \cdot 7x \equiv_{26} 15 \cdot 3 \equiv_{26} 19$.

Step 3. State the full set of solutions

So, the solutions are $19 + 26k$ for any $k \in \mathbb{Z}$

(must be of the form $a + mk$ for all $k \in \mathbb{Z}$ with $0 \leq a < m$)

Math mod a prime is especially nice

$\gcd(a, m) = 1$ if m is prime and $0 < a < m$ so can always solve these equations mod a prime.

+	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

x	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6
2	0	2	4	6	1	3	5
3	0	3	6	2	5	1	4
4	0	4	1	5	2	6	3
5	0	5	3	1	6	4	2
6	0	6	5	4	3	2	1

mod 7

Multiplicative Inverses and Algebra

Adding to both sides easily reversible:

$-c$ $\qquad$ $+c$

$$x \equiv_m y$$

$$x + c \equiv_m y + c$$

The same is not true of multiplication…

unless we have a multiplicative inverse $cd \equiv_m 1$

$\times\, d$ $\qquad$ $\times\, c$

$$x \equiv_m y$$

$$cx \equiv_m cy$$

Questions?

Modular Exponentiation mod 7

x	1	2	3	4	5	6
1	1	2	3	4	5	6
2	2	4	6	1	3	5
3	3	6	2	5	1	4
4	4	1	5	2	6	3
5	5	3	1	6	4	2
6	6	5	4	3	2	1

a	a^1	a^2	a^3	a^4	a^5	a^6
1	1	1	1	1	1	1
2	2	4	1	2	4	1
3	3	2	6	4	5	1
4	4	2	1	4	2	1
5	5	4	6	2	3	1
6	6	1	6	1	6	1

Exponentiation

- **Compute** 78365^{81453}

- **Compute** $78365^{81453} \bmod 104729$

- **Output is small**
  - need to keep intermediate results small

Small Multiplications

Since $b = qm + (b \bmod m)$, we have $b \bmod m \equiv_m b$.

And since $c = tm + (c \bmod m)$, we have $c \bmod m \equiv_m c$.

Multiplying these gives $(b \bmod m)(c \bmod m) \equiv_m bc$.

By the Lemma from a few lectures ago, this tells us
$bc \bmod m = (b \bmod m)(c \bmod m) \bmod m$.

Okay to mod b and c by m before multiplying if we are planning to mod the result by m

Repeated Squaring – small and fast

Since $b \bmod m \equiv_m b$ and $c \bmod m \equiv_m c$
we have $bc \bmod m = (b \bmod m)(c \bmod m) \bmod m$

So $a^2 \bmod m = (a \bmod m)^2 \bmod m$
and $a^4 \bmod m = (a^2 \bmod m)^2 \bmod m$
and $a^8 \bmod m = (a^4 \bmod m)^2 \bmod m$
and $a^{16} \bmod m = (a^8 \bmod m)^2 \bmod m$
and $a^{32} \bmod m = (a^{16} \bmod m)^2 \bmod m$

Can compute $a^k \bmod m$ for $k = 2^i$ in only i steps
What if k is not a power of 2?

Fast Exponentiation Algorithm

81453 in binary is 10011111000101101

$81453 = 2^{16} + 2^{13} + 2^{12} + 2^{11} + 2^{10} + 2^{9} + 2^{5} + 2^{3} + 2^{2} + 2^{0}$

$a^{81453} = a^{2^{16}} \cdot a^{2^{13}} \cdot a^{2^{12}} \cdot a^{2^{11}} \cdot a^{2^{10}} \cdot a^{2^{9}} \cdot a^{2^{5}} \cdot a^{2^{3}} \cdot a^{2^{2}} \cdot a^{2^{0}}$

a^{81453} mod m=

(…(((((a$^{2^{16}}$ mod m ·

a$^{2^{13}}$ mod m) mod m ·

a$^{2^{12}}$ mod m) mod m ·

a$^{2^{11}}$ mod m) mod m ·

a$^{2^{10}}$ mod m) mod m ·

a$^{2^{9}}$ mod m) mod m ·

a$^{2^{5}}$ mod m) mod m ·

a$^{2^{3}}$ mod m) mod m ·

a$^{2^{2}}$ mod m) mod m ·

a$^{2^{0}}$ mod m) mod m

Uses only 16 + 9 = 25 multiplications

The fast exponentiation algorithm computes $a^k \bmod m$ using $\leq 2\log k$ multiplications $\bmod m$

Fast Exponentiation: $a^k \bmod m$ for all k

Another way....

$$a^{2j} \bmod m = \left(a^j \bmod m\right)^2 \bmod m$$

$$a^{2j+1} \bmod m = \left((a \bmod m) \cdot (a^{2j} \bmod m)\right) \bmod m$$

Fast Exponentiation

```
public static int FastModExp(int a, int k, int modulus) {

        if (k == 0) {
                        return 1;

        } else if ((k % 2) == 0) {
                        long temp = FastModExp(a,k/2,modulus);
                        return (temp * temp) % modulus;

                } else {
                        long temp = FastModExp(a,k-1,modulus);
                        return (a * temp) % modulus;
                }
}
```

$$a^{2j} \bmod m = \left(a^{j} \bmod m\right)^{2} \bmod m$$

$$a^{2j+1} \bmod m = \left((a \bmod m) \cdot (a^{2j} \bmod m)\right) \bmod m$$

Using Fast Modular Exponentiation

- Your e-commerce web transactions use SSL (Secure Socket Layer) based on RSA encryption
- RSA
  - Vendor chooses random 512-bit or 1024-bit primes $\boldsymbol{p}, \boldsymbol{q}$ and 512/1024-bit exponent $\boldsymbol{e}$. Computes $\boldsymbol{m} = \boldsymbol{p} \cdot \boldsymbol{q}$
  - Vendor broadcasts $(\boldsymbol{m}, \boldsymbol{e})$
  - To send $\boldsymbol{a}$ to vendor, you compute $\boldsymbol{C} = \boldsymbol{a}^{\boldsymbol{e}} \bmod \boldsymbol{m}$ using *fast modular exponentiation* and send $\boldsymbol{C}$ to the vendor.
  - Using secret $\boldsymbol{p}, \boldsymbol{q}$ the vendor computes $\boldsymbol{d}$ that is the *multiplicative inverse* of $\boldsymbol{e}$ mod $(\boldsymbol{p} - \mathbf{1})(\boldsymbol{q} - \mathbf{1})$.
  - Vendor computes $\boldsymbol{C}^{\boldsymbol{d}} \bmod \boldsymbol{m}$ using *fast modular exponentiation*.
  - Fact: $\boldsymbol{a} = \boldsymbol{C}^{\boldsymbol{d}} \bmod \boldsymbol{m}$ for $\mathbf{0} < \boldsymbol{a} < \boldsymbol{m}$ unless $\boldsymbol{p} | \boldsymbol{a}$ or $\boldsymbol{q} | \boldsymbol{a}$