Number Theory I

↳ the study of integers.

Divisibility

a divides b, $a \mid b$, if there is an integer k such that $ak = b$.

e.g $7 \mid 63$ because $7 \cdot 9 = 63$.

Greek perfect number

- if equalled to the sum of its positive divisors, excluding itself

e.g $6 = 1 + 2 + 3$

$28 = 1 + 2 + 4 + 7 + 14$

* If $a|b$ and $b|c$, then $a|c$

a divides b, there exists an integer $k1$,

such that $ak_1 = b$

b divides c, there exists an integer $k2$,

such that $bk_2 = c$

, then: $ak_1 \cdot k_2 = c$

This implies $a|c$

* if $a|b$, then $a|bc$ for all c.

a divides b, $\forall\, k_1$ such that $ak_1 = b$

a divides bc, $\forall\, k_2$ such that $ak_2 = bc$

$$ak_2 = b \cdot c$$

$$ak_2 = ak_1 \cdot c$$

$\boxed{\text{for all } c}$

* if $a | b$ and $a | c$, then $a | sb + tc$ for all s & t

$a | b$, $\forall$ $ak_1 = b$

$a | c$, $\forall$ $ak_2 = c$

$a | sb + tc$, $\forall$ $ak_3 = sb + tc$

$$ak_3 = s(ak_1) + t(ak_2)$$

$$ak_3 = a(sk_1) + a(tk_2)$$

$$k_3 = (sk_1) + (tk_2)$$

a prime: a number $p > 1$ & $\not\exists$ positive divisors other than 1 & itself.

composite: every other number $> \underline{1}$

1 is considered $\not\exists$ prime nor composite.

Turing's code.

one approach:

- replace each letter of the message w/ two digits + append a number to make it a prime.

e.g A = 01 C = 03, etc

B = 02

" V I C T O R Y " (concatenate)

22 09 03 20 15 18 25 + 13

2209032015182513. is a prime

m = unencoded message.

m' = encrypted message.

Beforehand: the sender & receiver: a secret key, a large prime p.

Encryption: the sender encrypts the message m by computing

$$m' = m \cdot p$$

Decryption: the receiver decrypts m' by computing

$$\frac{m'}{p} = \frac{m \cdot p}{p} = m.$$

e.g: the secret key: 22801763489 → p

the message m = "victory" ⇒ m = "2209032015182513"

$$m' = m \cdot p$$
$$= 2209032015182513 * 22801763489$$

The Division algorithm

Let n and d be integers such that $d>0$,
Then $\forall$ a unique pair of integers q and r
Such that $n = qd + r$ and $0 \le r < d$.

Proof. $n = qd + r$ holds for some $r \geq 0$

If n is positive, the equation holds when
$q = 0$, $r = n$

$n = qd + r$

if n is not positive, then the equation holds when
$q = n$ and $r = n(1-d) \geq 0$.

Furthermore, r must be less than d,

otherwise, $b = (q+1) \cdot d + (r-d)$
would be another solution w/ a smaller
non negative remainder, contradicting the choice of R

$-11 = (-2) \cdot 7 + 3$

Note: $\underbrace{\boxed{\text{remainder}}}_{n = qd + r} \geq 0$ & $\boxed{\text{remainder}} < 7$.

Breaking Turing's code.

$$m_1' = m_1 \cdot p$$
$$m_2' = m_2 \cdot p$$

Note: after the p is recovered then every message can be read.

Modular arithmetic

→ a is congruent to b modulo c if $c \mid (a-b)$, denoted $a \equiv b \pmod{c}$

e.g: $29 \equiv 15 \pmod{7}$ because $7 \mid (29-15)$

congruent & Remainders

two numbers are congruent modulo c if and only if they have the same remainder when divided by c.

e.g: 19 and 32 are congruent modulo 13, because both have remainders of 6.

$a \equiv b \pmod{c}$ if and only if $(a \text{ rem } c) = (b \text{ rem } c)$

Proof $a \equiv b \pmod{c}$

$$(a \text{ rem } c) = (b \text{ rem } c)$$

By the division algorithm, there exist unique pairs of integers $q_1 r_1$ & $q_2 r_2$

1 $a = q_1 c + r_1 \quad (0 \leq r_1 < c)$

2 $b = q_2 c + r_2 \quad (0 \leq r_2 < c)$

$(a \text{ rem } c) = r_1$ & $(b \text{ rem } c) = r_2$

Substracting the second equation from the first

$$a - b = c(q_1 - q_2) + r_1 - r_2 \quad (-c < r_1 - r_2 < c)$$

$a \equiv b \pmod{c}$ if and only if c divides the left side.

This is true if and only if c divides the right side.

which holds if and only if $r_1 - r_2$ is a multiple of c

Given the bounds on $r_1 = r_2$, this happens precisely when $r_1 = r_2$, which is equivalent to

$$(a \text{ rem } c) = (b \text{ rem } c)$$

Note: Computer hardware works w/ fixed-sized chunks of data,

⇓

arbitrarily large integers in ordinary arithmetic are problematic

⇓

A standard solution:

- a computer w/ 64-bit internal registers typically does integer arithmetic modulo 2^{64}.

Thus an instruction to add the contents of registers A & B, actually computes $(A+B)$ rem 2^{64}.

Facts about rem & mod

$n \geq 1$, $a \equiv b \pmod n$ implies $a+c \equiv b+c \pmod n$.

$a \equiv b \pmod n$ means $n \mid (a-b)$

$a+c \equiv b+c \pmod n$ means $n \mid (a+c-b-c)$

$n \mid (a-b)$

Note: the difference between traditional vs modular arithmetic.

ordinary $a\cancel{c} = b\cancel{c}$ implies $a=b$ (provided $c \neq 0$)

$2 \cdot \cancel{3} \equiv 4 \cdot \cancel{3} \pmod 6$: False.

Lemma 27. The following assertions hold for all $n \geq 1$:

1). If $a_1 \equiv b_1 \pmod{n}$ and $a_2 \equiv b_2 \pmod{n}$ then $a_1 \cdot a_2 \equiv b_1 \cdot b_2 \pmod{n}$.

Proof: $n \mid a_1 - b_1$ and $n \mid a_2 - b_2$

$n \mid a_2(a_1 - b_1) + b_1(a_2 - b_2)$

note: $n \mid b$ and $n \mid c$ $n \mid sb + tc$ for all s and t

$n \mid a_1 a_2 - a_2 b_1 + a_2 b_1 - b_1 b_2$

$n \mid a_1 a_2 - b_1 b_2$

2). $(a \text{ rem } n) \equiv a \pmod{n}$

$a \text{ rem } n = a - qn$

$\boxed{a = qn + r}$

$n \mid qn$

$n \mid a - (a - qn)$

$n \mid a - (a \text{ rem } n)$

Note: $n \mid a - (a \text{ rem } n)$

$(a \text{ rem } n) \equiv a \pmod{n}$

3) $(a_1 \text{ rem } n) \cdot (a_2 \text{ rem } n) \cdots (a_k \text{ rem } n) \equiv a_1 \cdot a_2 \cdots a_k \pmod{n}$.

Turing's code (2.0)

Encryption:

The message m can be any integer in the set $\{1, 2, ... p-1\}$.

The sender encrypts the message m to produce m'

$$m' = mk \text{ rem } p.$$

Decryption: The receiver decripts m' by finding a message m to satisfy

$$m' = mk \text{ rem } p.$$

Cancellation Modulo a Prime.

Suppose p is a prime and k is not a multiple of p.

$$ak \equiv bk \pmod{p}$$

then

$$a \equiv b \pmod{p}.$$

Proof if $ak \equiv bk \pmod{p}$

$$p \mid (ak - bk)$$

$$p \mid k(a-b)$$

So p divides either k or $(a-b)$.

$p \, (a-b)$ means $a \equiv b \pmod{p}$

The relevance of $ak \equiv bk \pmod{p}$

$$\Downarrow$$

$a \equiv b \pmod{p}$

Since the messages a & b are drawn from the set $\{1, 2, \ldots p-1\}$

this means $a = b$

$$\Downarrow$$

two messages encrypt to the same thing only if they are themselves identical.

Note:

→ the encryption operation in Turing's code permutes the space of messages.

Corollary: suppose p is a prime & k is not a multiple of p

$(0 \cdot k)$ rem p, $(1 \cdot k)$ rem p, $(2 \cdot k)$ rem p, $\ldots ((p-1) \cdot k)$ rem p

is a permutation of the sequence:

$$0, 1, 2, \ldots (p-1).$$

e.g: $p = 5$ & $k = 3$

$(0 \cdot 3)$ rem 5 $= 0$, $(1 \cdot 3)$ rem 5 $= 3$, $(2 \cdot 3)$ rem 5 $= 1$, $(3 \cdot 3)$ rem 5 $= 4$, $(4 \cdot 3)$ rem 5 $= 2$

0, 3, 1, 4 is a permutation of 0, 1, 2, 3, 4.

Multiplicative Inverses.

The real numbers have a nice quality that the integers lack.

$\Downarrow$

non-zero real number r has a multiplicative inverse r^{-1}

, such that $r \cdot r^{-1} = 1$.

e.g.: multiplicative inverse of -3 is $-1/3$.

no integer can be multiplied by 5 to give 1.

$\Downarrow$

When we work modulo a prime number, p.

$\Downarrow$

most integers do have multiplicative inverses

, e.g.: we're working modulo 11.

, then the multiplicative inverse of 5 is 9.

$$5 \cdot 9 \equiv 1 \pmod{11}$$

The only exceptions are multiples of the modulus p

$\Downarrow$

$\emptyset$ inverses in the same way as 0 lacks an inverse in the real numbers.

Corollary 30,

Let p be a prime. if k is not a multiple of p, then there exists an integer $k^{-1} \in \{1, 2, p-1\}$

$$k \cdot k^{-1} \equiv 1 \pmod{p}.$$

so to decrypt

$$m' \cdot k^{-1} \text{ rem } p \equiv m' \cdot k^{-1} \pmod{p}$$

$$\equiv (mk \text{ rem } p) \cdot k^{-1} \pmod{p}$$

$$= mkk^{-1} \pmod{p}$$

$$\equiv m \pmod{p}$$

How to compute the k^{-1} (the multiplicative inverse).

Fermat's Theorem.

Suppose p is a prime and k is not a multiple of p.

$$k^{p-1} \equiv 1 \pmod p.$$

Proof.

$$1\cdot 2\cdot 3 \ldots (p-1) \equiv k(\text{rem } p).(2k \text{ rem } p).(3k \text{ rem } p).$$
$$((p-1)\cdot k \text{ rem } p) \pmod p$$
$$= k\cdot 2k\cdot 3k \ldots (p-1)k \pmod p$$
$$= (p-1)!\, k^{p-1} \pmod p.$$

$\Downarrow$

we cancle $(p-1)!$ as
p is a prime k does not divide
any of $1, 2, \ldots (p-1)$.

Multiplicative Inverse

$$k^{p-2}\cdot k \equiv 1 \pmod p.$$

k^{p-2} is a multiplicative inverse of k.

$\Downarrow$

e.g:

Compute: the multiplicative inverse of 6 modulo 17

$\Downarrow$

6^{15} rem $17 = 3$, so 3 is the multiplicative inverse

Finding inverse w/ fermat theorem.

$$k^{p-2} \cdot k \equiv 1.$$

k^{p-2} is a multiplicative inverse of k.

e.g: $x_1 \cdot 6 \equiv 1 \pmod{17}$

$\Rightarrow$ find x_1, a multiplicative inverse of 6 mod 17

Then we need to compute 6^{15} rem 17

Successive squaring:

$$6^{15} = 6^8 \cdot 6^4 \cdot 6^2 \cdot 6$$

$$6^2 = 36 \equiv 2$$
$$6^4 = (6^2)^2 \equiv 4$$
$$6^8 = (6^4)^2 \equiv 16$$

$$\equiv 16 \cdot 4 \cdot 2 \cdot 6 \equiv \boxed{3}$$