The Fundamentals: Algorithms, Integers, and Matrices

CSC-2259 Discrete Structures

The Growth of Functions

$f : R \rightarrow R$ $g : R \rightarrow R$

Big-Oh: $f(x)$ is $O(g(x))$

is no larger order than

Big-Omega: $f(x)$ is $\Omega(g(x))$

is no smaller order than

Big-Theta: $f(x)$ is $\Theta(g(x))$

is of same order as

Big-Oh: $f(x)$ is $O(g(x))$

(Notation abuse: $f(x) = O(g(x))$)

There are constants C, k (called witnesses) such that for all $x > k$:

$$|f(x)| \leq C \cdot |g(x)|$$

$$f(x) = x^2 \qquad g(x) = x^2 + 2x + 1$$

$$f(x) = O(g(x))$$

$$x^2 = O(x^2 + 2x + 1)$$

For $x > 0$: $x^2 \leq x^2 + 2x + 1$

$$f(x) \leq g(x)$$

Witnesses: $C = 1, \quad k = 0$

$f(x) = x^2 \qquad\qquad g(x) = x^2 + 2x + 1$

$$g(x) = O(f(x))$$

$$x^2 + 2x + 1 = O(x^2)$$

For $x > 1$: $\quad x^2 + 2x + 1 \leq x^2 + 2x^2 + x^2 = 4x^2$

$$g(x) \leq 4 \cdot f(x)$$

Witnesses: $C = 4, \quad k = 1$

$f(x) = O(g(x))$ and $g(x) = O(f(x))$

⇓

f and g are of the same order

Example: x^2 and $x^2 + 2x + 1$
are of the same order

$$f(x) = O(g(x)) \quad \text{and} \quad |g(x)| \leq |h(x)|$$

$$\Downarrow$$

$$f(x) = O(h(x))$$

Example: $$\left.\begin{array}{r} x^2 + 2x + 1 = O(x^2) \\ |x^2| \leq |x^3| \end{array}\right\} x^2 + 2x + 1 = O(x^3)$$

$$n^2 \neq O(n)$$

Suppose $n^2 = O(n)$

Then for all $n > k$: $$|n^2| \leq C \cdot |n|$$

$$\Downarrow$$

$$|n| \leq C$$

Impossible for $n > \max(C, k)$

Theorem: If $f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$

then $f(x) = O(x^n)$

Proof: for $x > 1$

$$\begin{aligned} |f(x)| &= |a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0| \\ &\le |a_n| x^n + |a_{n-1}| x^{n-1} + \cdots + |a_1| x + |a_0| \\ &\le |a_n| x^n + |a_{n-1}| x^n + \cdots + |a_1| x^n + |a_0| x^n \\ &= x^n (|a_n| + |a_{n-1}| + \cdots + |a_1| + |a_0|) \end{aligned}$$

Witnesses: $C = |a_n| + |a_{n-1}| + \cdots + |a_0|, \quad k = 1$

End of Proof

$$1 + 2 + \cdots + n = O(n^2)$$

$$1 + 2 + \cdots + n \le n + n + \cdots + n = n^2$$

Witnesses: $C = 1, \quad k = 1$

$$n! = 1 \cdot 2 \cdot \cdots \cdot n = O(n^n)$$

$$n! = 1 \cdot 2 \cdot \cdots \cdot n \leq n \cdot n \cdot \cdots \cdot n = n^n$$

Witnesses: $C = 1, \quad k = 1$

$$2^n = O(n!)$$

$$\begin{aligned} 2^n &= 2 \cdot 2^{n-1} \\ &= 2 \cdot (2 \cdot 2 \cdots \cdot 2) \\ &\leq 2 \cdot (2 \cdot 3 \cdots \cdot n) \\ &= 2 \cdot n! \end{aligned}$$

Witnesses: $C = 2, \quad k = 2$

$$\log n! = O(n \cdot \log n)$$

$$\log n! \leq \log n^n = n \cdot \log n$$

Witnesses: $C = 1, \quad k = 1$

$$n = O(2^n)$$

$$\log n = O(n)$$

For $n > 1$: $\quad n < 2^n$

$$\Downarrow$$

$$\log n < \log 2^n = n \cdot \log 2 = n$$

Witnesses: $C = 1, \quad k = 1$

$$\log_a n = O(\log n)$$

$$\log_a n = \frac{\log n}{\log a}$$

Witnesses: $C = \frac{1}{\log a}, \quad k = 1$

constant $\frac{1}{x} = O(1)$

For $x > 1$: $\frac{1}{x} \leq 1$

Witnesses: $C = 1, \quad k = 1$

Interesting functions

$$1 \quad \log n \quad n \quad n\log n \quad n^2 \quad 2^n \quad n!$$

$$\longrightarrow$$

Higher growth

Theorem: If $f_1(x) = O(g_1(x))$, $f_2(x) = O(g_2(x))$

then $(f_1 + f_2)(x) = O(\max(|g_1(x)|,|g_2(x)|))$

Proof: $x > k_1 \qquad |f_1(x)| \leq C_1 \cdot |g_1(x)|$

$x > k_2 \qquad |f_2(x)| \leq C_2 \cdot |g_2(x)|$

$$x > \max(k_1, k_2) \qquad \begin{aligned} &|(f_1 + f_2)(x)| = |f_1(x) + f_2(x)| \leq |f_1(x)| + |f_2(x)| \\ &\leq C_1 |g_1(x)| + C_2 |g_2(x)| \\ &\leq (C_1 + C_2) \cdot \max(|g_1(x)|,|g_2(x)|) \end{aligned}$$

Witnesses: $C = C_1 + C_2, \quad k = \max(k_1, k_2)$

End of Proof

Corollary: If $f_1(x) = O(g(x))$, $f_2(x) = O(g(x))$

then $(f_1 + f_2)(x) = O(g(x))$

Theorem: If $f_1(x) = O(g_1(x))$, $f_2(x) = O(g_2(x))$

then $(f_1 f_2)(x) = O(g_1(x) g_2(x))$

$$3n\log(n!) + (n^2+3)\log n = O(n^2 \log n)$$

Multiplication

$$\left.\begin{array}{r} 3n = O(n) \\ \log(n!) = O(n\log n) \end{array}\right\} \begin{array}{l} 3n\log(n!) \\ = O(n \cdot n \log n) \\ = O(n^2 \log n) \end{array}$$

$$\left.\begin{array}{r} n^2 + 3 = O(n^2) \\ \log n = O(\log n) \end{array}\right\} \begin{array}{l} (n^2+3)\log(n) \\ = O(n^2 \log n) \end{array}$$

Addition

$$\begin{array}{l} 3n\log(n!) + (n^2+3)\log n \\ = O(n^2 \log n) \end{array}$$

Big-Omega: $f(x)$ is $\Omega(g(x))$

(Notation abuse: $f(x) = \Omega(g(x))$)

There are constants C, k (called witnesses) such that for all $x > k$:

$$|f(x)| \geq C \cdot |g(x)|$$

$$8x^3 + 5x^2 + 7 = \Omega(x^3)$$

$x > 1$ $\qquad 8x^3 + 5x^2 + 7 \geq 8x^3$

Witnesses: $C = 8, \quad k = 1$

Same order

Big-Theta: $f(x)$ is $\Theta(g(x))$

(Notation abuse: $f(x)=\Theta(g(x))$)

$$f(x)=O(g(x)) \text{ and } f(x)=\Omega(g(x))$$

Alternative definition:

$$f(x)=O(g(x)) \text{ and } g(x)=O(f(x))$$

$$3x^2+8x\log x=\Theta(x^2)$$

$$3x^2+8x\log x\le 3x^2+8x^2=11x^2$$

$3x^2+8x\log x=O(x^2)$ Witnesses: $C=11,\quad k=1$

$$3x^2+8x\log x\ge 3x^2$$

$3x^2+8x\log x=\Omega(x^2)$ Witnesses: $C=3,\quad k=1$

Theorem: If $f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$

then $f(x) = \Theta(x^n)$

Proof: We have shown: $f(x) = O(x^n)$

We only need to show $f(x) = \Omega(x^n)$

Take $x > 1$ and examine two cases

Case 1: $a_n > 0$

Case 2: $a_n < 0$

Case 1: $a_n > 0$ $(x > 1)$

$$b = \max(|a_{n-1}|, |a_{n-2}|, \ldots, |a_0|)$$

$$\begin{aligned} f(x) &= a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 \\ &\geq a_n x^n - nbx^{n-1} \\ &\geq a' x^n \end{aligned}$$

For $0 < a' < a_n$ and $x \geq \dfrac{nb}{(a_n - a')}$

Case 2 is similar

End of Proof

Complexity of Algorithms

Time complexity

Number of operations performed

Space complexity

Size of memory used

Linear search algorithm

```
Linear-Search( $x$,  $a_1, a_2, \ldots, a_n$) {
    $i \leftarrow 1$
  while( $i \leq n$ and $x \neq a_i$)
    $i++$
  if ( $i \leq n$) return $i$        //item found
  else return $0$                  //item not found
}
```

Time complexity

Comparisons

Item not found in list: $2(n+1)+1$

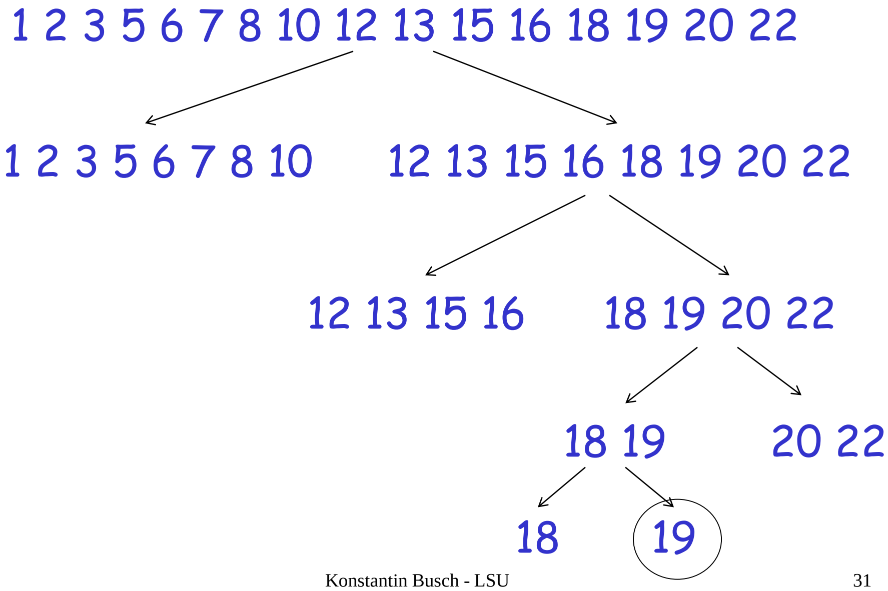
Item found in position i: $2i+1$

Worst case performance: $2(n+1)+1=O(n)$

Binary search algorithm

```
Binary-Search( $x,\quad a_1, a_2, \ldots, a_n$ ) {
  $i \leftarrow 1$        //left endpoint of search area
  $j \leftarrow n$        //right endpoint of search area
  while($i <$ j ) {
     $m \leftarrow \lfloor (i+j)/2 \rfloor$
     if ($x > \mathrm{a_m}$) $i \leftarrow m+1$   //item is in right half
     else $j \leftarrow m$                  //item is in left half
  }
  if ($x = \mathrm{a_i}$ ) return $i$     //item found
  else return $0$              //item not found
}
```

Search 19



1 2 3 5 6 7 8 10 12 13 15 16 18 19 20 22

1 2 3 5 6 7 8 10

12 13 15 16 18 19 20 22

12 13 15 16

18 19 20 22

18 19

20 22

18

19

Time complexity

Size of search list at iteration 1: $\frac{n}{2^0}$

Size of search list at iteration 2: $\frac{n}{2^1}$

$\vdots$

Size of search list at iteration k: $\frac{n}{2^{k-1}}$

Size of search list at iteration k: $\frac{n}{2^{k-1}}$

Smallest list size: 1

in last iteration m: $\frac{n}{2^{m-1}} = 1$

$$\Downarrow$$

$$m = 1 + \log n$$

Total comparisons:

$$\underbrace{(1+\log n)}_{\text{\#iterations}} \cdot \underbrace{2}_{\text{Comparisons per iteration}} + \underbrace{1}_{\text{Last comparison}} = \Theta(\log n)$$

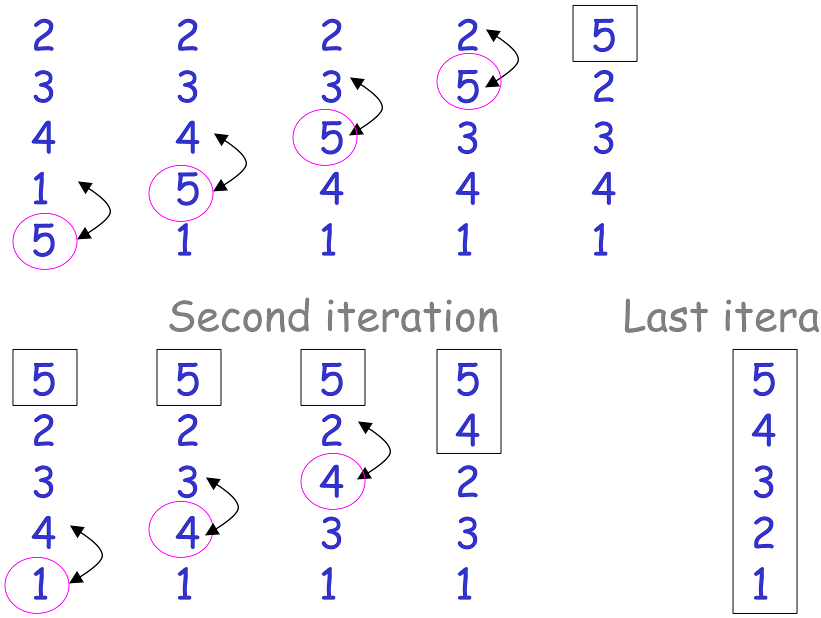
Bubble sort algorithm

```
Bubble-Sort( $a_1, a_2, \ldots, a_n$ ) {
  for ( $i \leftarrow 1$ to $n-1$ ) {
    for ( $j \leftarrow 1$ to $n-i$ )
      if ( $a_j > a_{j+1}$ )
        swap $a_j, a_{j+1}$
}
```

First iteration

Second iteration

Last iteration



First iteration

2	2	2	2	5
3	3	3	5	2
4	4	5	3	3
1	5	4	4	4
5	1	1	1	1

Second iteration

5	5	5	5
2	2	2	4
3	3	4	2
4	4	3	3
1	1	1	1

Last iteration

5
4
3
2
1

Time complexity

Comparisons in iteration 1: $n-1$

Comparisons in iteration 2: $n-2$

$\vdots$

Comparisons in iteration $n-1$: 1

Total: $1+2+\cdots+(n-1)=\dfrac{(n-1)n}{2}=\Theta(n^2)$

Tractable problems

Class P:

Problems with algorithms whose time complexity is polynomial $O(n^b)$

Examples: Search, Sorting, Shortest path

Intractable problems

Class NP :

Solution can be verified in polynomial time but no polynomial time algorithm is known

Examples: Satisfiability, TSP, Vertex coloring

Important computer science question

$$P = NP\ ?$$

Unsolvable problems

There exist unsolvable problems which do not have any algorithm

Example: Halting problem in Turing Machines

Integers and Algorithms

Base b expansion of integer n:

$$n = a_k b^k + a_{k-1} b^{k-1} + \cdots + a_1 b^1 + a_0$$

$$(a_k a_{k-1} \cdots a_1 a_0)_b$$

Integers: $k \geq 0$ $\quad 0 \leq a_i < b$

Example: $(276)_{10} = 2 \cdot 10^2 + 7 \cdot 10 + 6$

Binary expansion

Digits: $0,1$

$$(1\,01011111)_2$$

$$= 1 \cdot 2^8 + 0 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5$$

$$+ 1 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$$

$$= 351$$

Hexadecimal expansion

Digits: $0,1,2,\ldots,9,A,B,C,D,E,F$

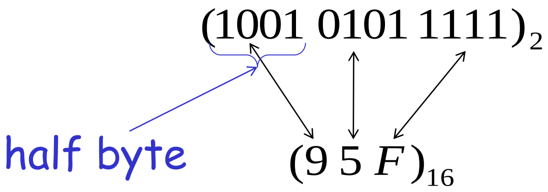
$$
\begin{aligned}
&(2AE0B)_{16} \\
&= 2\cdot 16^4 + 10\cdot 16^3 + 14\cdot 16^2 + 0\cdot 16 + 11 \\
&= 175627
\end{aligned}
$$

Octal expansion

Digits: $0,1,2,\ldots,7$

$$
\begin{aligned}
&(245)_8 \\
&= 2\cdot 8^2 + 4\cdot 8 + 5 \\
&= 165
\end{aligned}
$$

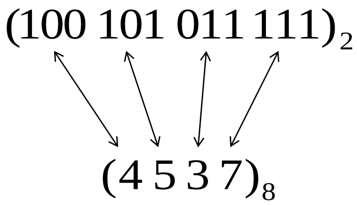
Conversion between binary and hexadecimal


$(1001\ 0101\ 1111)_2$

half byte

$(9\ 5\ F)_{16}$

Conversion between binary and octal


$(100\ 101\ 011\ 111)_2$

$(4\ 5\ 3\ 7)_8$

```
Base $b$ expansion( $n$ ) {
  $q \leftarrow n$
  $k \leftarrow 0$
  While ( $q \neq 0$ ) {
      $a_k \leftarrow q \bmod b$
      $q \leftarrow \lfloor q / b \rfloor$
      $k \leftarrow k + 1$
  }
  return $(a_{k-1} a_{k-2} \cdots a_1 a_0)_b$
}
```

Binary expansion of $241 = (11110001)_2$

$$
\begin{array}{ccccccc}
241 & = & 2\cdot 120 & + & 1 & & a_0 \\
120 & = & 2\cdot 60 & + & 0 & & a_1 \\
60 & = & 2\cdot 30 & + & 0 & & a_2 \\
30 & = & 2\cdot 15 & + & 0 & & \\
15 & = & 2\cdot 7 & + & 1 & & \vdots \\
7 & = & 2\cdot 3 & + & 1 & & \\
3 & = & 2\cdot 1 & + & 1 & & \\
1 & = & 2\cdot 0 & + & 1 & & a_7
\end{array}
$$

Octal expansion of $12345 = (30071)_8$

$$
\begin{array}{ccccccc}
12345 & = & 8\cdot 1543 & + & 1 & & a_0 \\
1543 & = & 8\cdot 192 & + & 7 & & a_1 \\
192 & = & 8\cdot 24 & + & 0 & & a_2 \\
24 & = & 8\cdot 3 & + & 0 & & a_3 \\
3 & = & 8\cdot 0 & + & 3 & & a_4
\end{array}
$$

```
Binary_addition($a, b$) {
    $a = (a_{n-1}a_{n-2}\cdots a_1a_0)_2$
    $b = (b_{n-1}b_{n-2}\cdots b_1b_0)_2$
    $c \leftarrow 0$                    //carry bit
    for $j \leftarrow 0$ to $n-1$ {
        $d \leftarrow \lfloor (a_j + b_j + c)/2 \rfloor$ //auxilliary
        $s_j \leftarrow a_j + b_j + c - 2d$   //j sum bit
        $c \leftarrow d$                //carry bit
    }
    $s_n \leftarrow c$                  //last sum bit
    return $(s_ns_{n-1}\cdots s_1s_0)_2$
}
```

Carry bit: 1 1 1

$$\begin{array}{rl} 1110 & a \\ +1011 & b \\ \hline 11001 & \end{array}$$

Time complexity of binary addition: $O(n)$

(counting bit additions) $O(\log a)$

```
Binary_multiplication($a, b$) {
    $a = (a_{n-1}a_{n-2}\cdots a_1a_0)_2$
    $b = (b_{n-1}b_{n-2}\cdots b_1b_0)_2$
    for $j \leftarrow 0$ to $n-1$ {
        if ($b_j = 1$)
            $c_j \leftarrow a \cdot 2^j$    //a shifted j positions
        else
            $c_j \leftarrow 0$
    }
    $p \leftarrow c_0 + c_1 + \cdots + c_{n-1}$
    return binary expansion of $p$
}
```

$$\begin{array}{cccccccc}
 & & & 1 & 1 & 0 & & a \\
 & & \times & 1 & 0 & 1 & & b \\
\hline
 & & & 1 & 1 & 0 & & c_0 \\
 & & 0 & 0 & 0 & & & c_1 \\
+ & 1 & 1 & 0 & & & & c_2 \\
\hline
 & 1 & 1 & 1 & 1 & 0 & &
\end{array}$$

Time complexity of multiplication: $O(n)$

(counting shifts and bit additions) $O(\log^2 a)$

Integers and Division

Integers $a, b \quad (a \neq 0)$

a divides b: $\quad a \mid b \qquad \exists c, \ b = a \cdot c$

factor

Examples: $\quad 3 \mid 12 \qquad 12 = 3 \cdot 4$

$3 \nmid 7$

a, b, c integers

if $a \mid b$ then $a \mid bc$

$$a \mid b \Rightarrow \exists s \quad b = a \cdot s \Rightarrow bc = a \cdot (sc)$$

a, b, c integers

if $a \mid b$ and $a \mid c$ then $a \mid (b+c)$

$$\left.\begin{array}{lll} a \mid b & \Rightarrow \exists s & b = a \cdot s \\ a \mid c & \Rightarrow \exists t & c = a \cdot t \end{array}\right\} \; b + c = a \cdot (s+t)$$

a, b, c integers

if $a \mid b$ and $b \mid c$ then $a \mid c$

$$\left.\begin{array}{lll} a \mid b & \Rightarrow \exists s & b = a \cdot s \\ b \mid c & \Rightarrow \exists t & c = b \cdot t \end{array}\right\} \; c = a \cdot st$$

a, b, c, m, n integers

if $a \mid b$ and $a \mid c$ then $a \mid mb + nc$

$$\left.\begin{array}{l} a \mid b \Rightarrow a \mid mb \\ a \mid c \Rightarrow a \mid nc \end{array}\right\} \Rightarrow a \mid mb + nc$$

The division "algorithm"

$a \in Z \qquad d \in Z^+$

There are unique $q, r \in Z$ such that:

$$a = \underset{\text{divisor}}{d} \cdot \underset{\text{quotient}}{q} + \underset{\text{remainder}}{r}$$

$$0 \le r < d$$

$$a = d \cdot q + r$$

$$q = a \ \text{ div } \ d \qquad\qquad r = a \ \text{ mod } \ d$$

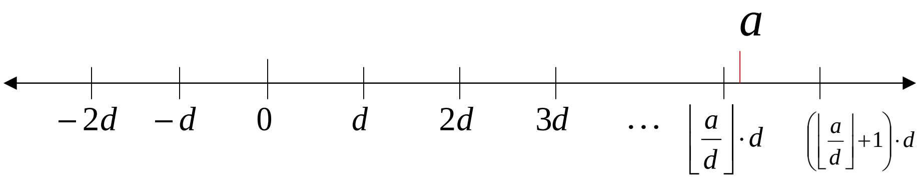
$$q = \left\lfloor \frac{a}{d} \right\rfloor \qquad\qquad r = \left| a - \left\lfloor \frac{a}{d} \right\rfloor d \right|$$

Examples: $101 = 11 \cdot 9 + 2$

$$9 = 101 \ \text{ div } \ 11 \qquad 2 = 101 \ \text{ mod } \ 11$$

$$-11 = 3(-4) + 1$$

$$-4 = -11 \ \text{ div } \ 3 \qquad 1 = -11 \ \text{ mod } \ 3$$


a
$-2d$
$-d$
0
d
$2d$
$3d$
$\ldots$
$\left\lfloor \frac{a}{d} \right\rfloor \cdot d$
$\left(\left\lfloor \frac{a}{d} \right\rfloor + 1 \right) \cdot d$

Number of positive integers divisible by d and not exceeding a:

$$\left\lfloor \frac{a}{d} \right\rfloor$$

```
Division_algorithm($a, d$) {
    $q \leftarrow 0$    $r \leftarrow |a|$
    while ($r \geq d$) {
        $r \leftarrow r - d$
        $q \leftarrow q + 1$
    }
    if ($a < 0$ and $r > 0$) {   //a is negative
        $r \leftarrow d - r$            //adjust r
        $q \leftarrow -(q+1)$          //adjust q
    }
    else if ($a < 0$) { $q \leftarrow -q$ }
    return  $q$ ($a$ div $d$) , $r$ ($a$ mod $d$)
}
```

$a = 15$

$d = 4$

r	q
15	0
$15 - 4 = 11$	1
$11 - 4 = 7$	2
$7 - 4 = 3$	3

$r = 15 \bmod 4 = 3$ $q = 15 \text{ div } 4 = 3$

Time complexity of division alg.: $O(q \log a)$

There is a better algorithm: $O(\log a \cdot \log d)$
(based on binary search)

Modular Arithmetic

$a, b \in Z$ $\qquad$ $m \in Z^{+}$

$$a \equiv b \pmod{m}$$

"a is congruent to b modulo m"

$$a \bmod m = b \bmod m$$

Examples: $1 \equiv 13 \pmod{12}$ $\qquad$ $0 \equiv m \pmod{m}$

$11 \equiv 5 \pmod{6}$ $\qquad$ $k \cdot m \equiv 0 \pmod{m}$

Equivalent definitions

$$a \equiv b \pmod{m}$$

$$\Updownarrow$$

$$a \bmod m = b \bmod m$$

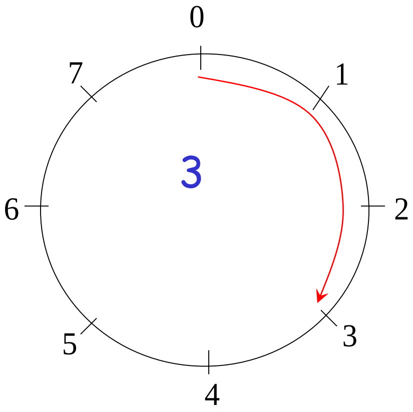
$$\Updownarrow$$

$$m \mid a - b$$

$$\Updownarrow$$

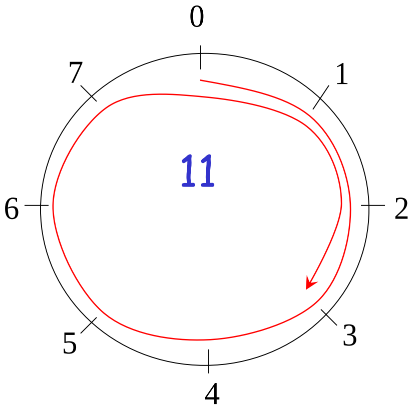
$$\exists k \in Z, \quad a = b + km$$

$$3 \bmod 8 = 3$$



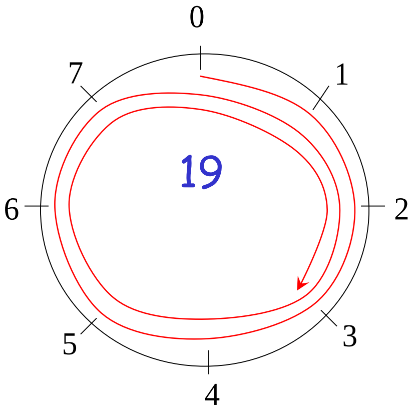
Length of line represents number

$$11 \bmod 8 = 3$$



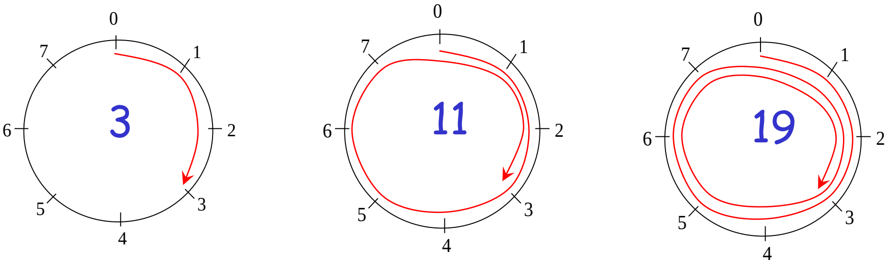
Length of helix line represents number

$$19 \bmod 8 = 3$$



Length of helix line represents number

$$3 \equiv 11 \equiv 19 \pmod{8}$$



Helix lines terminate in same number

Congruence class of a modulo m:

$$S_a = \{b \mid a \equiv b \pmod{m}\}$$

There are m congruence classes:

$$S_0, S_1, \ldots, S_{m-1}$$

$$\left.\begin{array}{l} a \equiv b \pmod{m} \\ c \equiv d \pmod{m} \end{array}\right\} \Rightarrow a + c \equiv b + d \pmod{m}$$

$$\left.\begin{array}{lll} a \equiv b \pmod{m} & \Rightarrow & a = b + sm \\ c \equiv d \pmod{m} & \Rightarrow & c = d + tm \end{array}\right\} a + c = d + b + (s + t)m$$

$$\left.\begin{array}{l} a \equiv b \pmod{m} \\ \\ c \equiv d \pmod{m} \end{array}\right\} \Rightarrow \; a \cdot c \equiv b \cdot d \pmod{m}$$

$$\left.\begin{array}{l} a \equiv b \pmod{m} \Rightarrow a = b + sm \\ \\ c \equiv d \pmod{m} \Rightarrow c = d + tm \end{array}\right\} \begin{array}{l} a \cdot c = (b + sm)(d + tm) \\ \qquad = \mathrm{bd} + m(bt + ds + stm) \end{array}$$

$$7 \equiv 2 \pmod{5}$$

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

$$18 = 7 + 11 \equiv (2 + 1) \pmod{5} = 3 \pmod{5}$$

$$77 = 7 \cdot 11 \equiv (2 \cdot 1) \pmod{5} = 2 \pmod{5}$$

$$(a+b) \bmod m = ((a \bmod m) + (b \bmod m)) \bmod m$$

$$ab \bmod m = ((a \bmod m)(b \bmod m)) \bmod m$$

Follows from previous results by using:

$$a \bmod m = (a \bmod m) \bmod m$$

$$b \bmod m = (b \bmod m) \bmod m$$

Modular exponentiation

Compute $b^n \bmod m$ efficiently using small numbers

$$b^n = b^{\overbrace{a_{k-1}2^{k-1}+\cdots+a_1 2+a_0}^{\text{Binary expansion of } n}} = b^{a_{k-1}2^{k-1}} \cdots b^{a_1 2} b^{a_0}$$

$$\begin{aligned} & b^n \bmod m \\ & = b^{a_{k-1}2^{k-1}} \cdots b^{a_1 2} b^{a_0} \bmod m \\ & = ((b^{a_{k-1}2^{k-1}} \bmod m) \cdot \cdots \cdot (b^{a_1 2} \bmod m) \cdot (b^{a_0} \bmod m)) \bmod m \end{aligned}$$

Example: $3^{644} \bmod 645 = 36$

$$644 = 1010000100 \; = 2^9 + 2^7 + 2^2$$

$$3^{644} = 3^{2^9+2^7+2^2} = 3^{2^9} 3^{2^7} 3^{2^2}$$

$$\begin{aligned}
&3^{644} \bmod 645 \\
&= (3^{2^9} 3^{2^7} 3^{2^2}) \bmod 645 \\
&= ((3^{2^9} \bmod 645)(3^{2^7} \bmod 645)(3^{2^2} \bmod 645) \bmod 645)
\end{aligned}$$

Compute all the powers of 3 efficiently

$$\begin{aligned}
&3^2 \bmod 645 = 9 \bmod 645 = 9 \\
&3^{2^2} \bmod 645 = \left(3^2\right)^2 \bmod 645 = ((3^2 \bmod 645)(3^2 \bmod 645)) \bmod 645 = (9 \cdot 9 \bmod 645) = 81 \\
&3^{2^3} \bmod 645 = \left(3^{2^2}\right)^2 \bmod 645 = ((3^{2^2} \bmod 645)(3^{2^2} \bmod 645)) \bmod 645 = 81 \cdot 81 \bmod 645 = 111 \\
&\vdots \\
&3^{2^9} \bmod 645 = \left(3^{2^8}\right)^2 \bmod 645 = ((3^{2^8} \bmod 645)(3^{2^8} \bmod 645)) \bmod 645 = 111
\end{aligned}$$

Use the powers of 3 to get result efficiently

$$\begin{aligned}
&3^{644} \\
&= (3^{2^9} 3^{2^7} 3^{2^2} \bmod 645) \\
&= (3^{2^9} 3^{2^7} (3^{2^2} \bmod 645) \bmod 645) = (3^{2^9} 3^{2^7} 81 \bmod 645) \\
&= (3^{2^9} (((3^{2^7} \bmod 645) 81) \bmod 645) \bmod 645) = (3^{2^9} ((396 \cdot 81) \bmod 645) \bmod 645) = (3^{2^9} \cdot 471 \bmod 645) \\
&= (((3^{2^9} \bmod 645) \cdot 471) \bmod 645) = 111 \cdot 471 \bmod 645 = 36
\end{aligned}$$

```
Modular_Exponentiation( $b, n, m$ ) {
    $n = (a_{n-1}a_{n-2}\cdots a_1 a_0)_2$
    $x \leftarrow 1$
    $power \leftarrow b \bmod m$
    for $i = 0$ to $k-1${
        if ($a_i = 1$) $x \leftarrow (x \cdot power) \bmod m$
        $power \leftarrow (power \cdot power) \bmod m$
    }
    return $x$    $(b^n \bmod m)$
}
```

Time complexity: $O(\log^2 m \cdot \log n)$

bit operations

Congruent application: Hashing functions

$$h(k) = k \bmod m$$

Example: $h(k) = k \bmod 111$

Employer id Folder#

$h(064212848) = 064212848 \bmod 111 = 14$

$h(037149212) = 037149212 \bmod 111 = 65$

$h(107405723) = 107405723 \bmod 111 = 14$ collision

Application: Pseudorandom numbers

Sequence of pseudorandom numbers

$$x_0, x_1, x_2, \ldots$$

Linear congruential method: $x_{n+1} = (ax_n + c) \bmod m$

$2 \le a < m$

$0 \le c < m$

seed

$0 \le x_0 < m$

seed

Example: $x_{n+1} = (7x_n + 4) \bmod 9$ $\quad x_0 = 3$

$\boxed{3,7,8,6,1,2,0,4,5},\boxed{3,7,8,6,1,2,0,4,5},3\ldots$

Application: Cryptology

$$\begin{tikzcd}
\text{"MEET YOU IN THE PARK"} \arrow[d, shift right=2, "{\text{encryption}\; f(x)=(x+3)\bmod m}"'] \\
\text{"PHHW BRX LQ WKH SDUN"} \arrow[u, shift right=2, "{\text{decryption}\; f^{-1}(x)=(x-3)\bmod m}"']
\end{tikzcd}$$

Shift cipher: $f(x) = (x+2) \bmod m$

Affine transformation: $f(x) = (ax+b) \bmod m$

Primes and Greatest Common Divisor

Prime p: Positive integer greater than 1, only positive factors are $1, p$

Non-prime = composite

Primes: 2,3,5,7,11,13,17,19,23,29,31,37,41,...

Fundamental theorem of arithmetic

Every positive integer is either prime or a unique product of primes

Prime factorization: $m = p_1^{k_1} p_2^{k_2} p_3^{k_3} \cdots p_l^{k_l}$

↗ prime

Examples: $100 = 2^5 \cdot 5^2$ $\quad 999 = 3^3 \cdot 37$

$7007 = 7^2 \cdot 11 \cdot 13$

Theorem: If n is composite then it has prime divisor $p \leq \sqrt{n}$

Proof:

n is composite $\Rightarrow$ $\exists a, \exists b, 1 < a, b < n, n = ab$

$c = \min(a,b) \leq \sqrt{n}$ since otherwise $ab > \sqrt{n}\sqrt{n} = n$

From fundamental theorem of arithmetic c is either prime or has a prime divisor

End of Proof

```
Prime_factorization( $n$ ) {
    $p \leftarrow 2$    //first prime
    $n' \leftarrow n$
    while ( $n' > 1$ and $p \leq \sqrt{n'}$ ) {
        if ( $p$ divides $n'$ ) {
            $p$ is a factor of $n$
            $n' \leftarrow n' / p$
        }
        else
            $p \leftarrow$ next prime after $p$
    }
    return all prime factors found
}
```

$$n = 7007$$

$p = 2,3,5$ do not divide 7007

$p = 7$ $\quad 7007 = 7 \cdot 1001$ $\quad n'$

$p = 7$ $\quad 1001 = 7 \cdot 143$

$p = 7$ $\quad$ does not divide 143

$p = 11$ $\quad 143 = 11 \cdot 13$

$p = 11$ $\quad 13 \quad (11 > \sqrt{13})$

$$n = 7 \cdot 7 \cdot 11 \cdot 13 = 7^2 \cdot 11 \cdot 13$$

Theorem: There are infinitely many primes

Proof: Suppose finite primes $p_1, p_2, \ldots, p_k$

Let $q = p_1 p_2 \cdots p_k + 1$

$$\left.\begin{array}{l}\text{If some prime } p_i \mid q \\ \text{Since } p_i \mid -p_1 p_2 \cdots p_k\end{array}\right\} \Rightarrow p_i \mid q - p_1 p_2 \cdots p_k = 1$$

impossible

No prime divides q $\Rightarrow$ q is prime

(From fundamental theorem of arithmetic)

Contradiction!

End of Proof

Largest prime known (as of 2006)

$$2^{30,402,457}-1$$

Mersenne primes have the form: 2^k-1

$2^2-1=3$ $2^3-1=5$ $2^5-1=31$

Prime number theorem

The number of primes less or equal to n approaches to:

$$\frac{n}{\ln n}$$

$\log_e n$

Goldbach's conjecture:

Every integer is the sum of two primes

$4 = 2 + 2 \qquad 6 = 3 + 3 \qquad 6 = 5 + 3 \qquad 10 = 7 + 3$

Twin prime conjecture:

There are infinitely many twin primes

Twin primes differ by 2: $3{,}5 \quad 5{,}7 \quad 11{,}13 \quad 17{,}19$

Greatest common divisor

$\gcd(a, b) =$ largest integer d such that $d \mid a$ and $d \mid b$

$a, b \in Z$

$|a| + |b| \neq 0$

Examples: $\gcd(24, 36) = 12$

Common divisors of 24, 36: 1, 2, 3, 4, 6, 12

$\gcd(17, 22) = 1$

Common divisors of 17, 22: 1

Trivial cases:

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

$$\gcd(m,0) = m \qquad\qquad m \neq 0$$

Theorem: If $a = b \cdot q + r$ $\qquad \overset{(a/b)}{0 \le r < b}$

then $\gcd(a,b) = \gcd(b,r)$

Proof:

$$\begin{matrix} d \mid a \\ d \mid b \end{matrix} \Rightarrow \begin{matrix} a = ds \\ b = dt \end{matrix} \Rightarrow \begin{matrix} r = d(s - tq) \\ b = dt \end{matrix} \Rightarrow \begin{matrix} d \mid r \\ d \mid b \end{matrix}$$

Thus, (a,b) and (b,r) have
the same set of common divisors

End of proof

divisions $a = r_0$ $b = r_1$ remainder

$$\begin{array}{lrcll}
r_0 / r_1 & r_0 & = & r_1 q_1 + r_2 & 0 < r_2 < r_1 \\
r_1 / r_2 & r_1 & = & r_2 q_2 + r_3 & 0 < r_3 < r_2 \\
\vdots & & \vdots & & \vdots \\
r_{n-2} / r_{n-1} & r_{n-2} & = & r_{n-1} q_{n-1} + r_n & 0 < r_n < r_{n-1} \\
r_{n-1} / r_n & r_{n-1} & = & r_n q_n + 0 &
\end{array}$$

first zero result

$$\gcd(a,b) = \gcd(r_0, r_1) = \gcd(r_1, r_2) = \gcd(r_2, r_3) \cdots$$

$$\cdots = \gcd(r_{n-2}, r_{n-1}) = \gcd(r_{n-1}, r_n) = \gcd(r_n, 0) = r_n$$

$a = 662$ $b = 414$

$$\begin{array}{rcll}
662 & = & 414 \cdot 1 + 248 & r_2 = 248 < 414 = r_1 \\
414 & = & 248 \cdot 1 + 166 & r_3 = 166 < 248 = r_2 \\
248 & = & 166 \cdot 1 + 82 & r_4 = 82 < 166 = r_3 \\
166 & = & 82 \cdot 2 + 2 & r_5 = 2 < r_4 = 82 \\
82 & = & 2 \cdot 41 + 0 &
\end{array}$$

result

$$\gcd(662,414) = \gcd(414,248) = \gcd(248,166)$$

$$= \gcd(166,82) = \gcd(82,2) = \gcd(2,0) = 2$$

$$\begin{array}{ccccccccc} a & b & & & & & & & \\ r_0 & r_1 & r_2 & r_3 & r_4 & \cdots & r_{n-1} & r_n & 0 \end{array}$$

$r_0 \bmod r_1 = r_2$

$r_1 \bmod r_2 = r_3$

$r_2 \bmod r_3 = r_4$

$r_{n-2} \bmod r_{n-1} = r_n$

$r_{n-1} \bmod r_n = 0$

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

$$\begin{array}{ccccccc} a & b & & & & r_n & \\ 662 & 414 & 248 & 166 & 82 & 2 & 0 \end{array}$$

$662 \bmod 414 = 248$

$414 \bmod 248 = 166$

$248 \bmod 166 = 82$

$166 \bmod 82 = 2$

$82 \bmod 2 = 0$

$$\gcd(a,b) = r_n = 2$$

Descending sequence:

$$\overset{a}{r_0} > \overset{b}{r_1} > \quad \cdots \quad > r_i > r_{i+1} > r_{i+2} > \quad \cdots \quad > r_n > 0$$

$$r_i \bmod r_{i+1} \;=\; r_{i+2}$$

Property: $\dfrac{r_i}{2} > r_{i+2}$

Case 1: $\dfrac{r_i}{2} \geq r_{i+1}$ $\Longrightarrow$ $\dfrac{r_i}{2} \geq r_{i+1} > r_{i+2}$

Descending sequence:

$$\overset{a}{r_0} > \overset{b}{r_1} > \quad \cdots \quad > r_i > r_{i+1} > r_{i+2} > \quad \cdots \quad > r_n > 0$$

$$r_i \bmod r_{i+1} \;=\; r_{i+2}$$

Property: $\dfrac{r_i}{2} > r_{i+2}$

Case 2: $\dfrac{r_i}{2} < r_{i+1}$ $\Longrightarrow$ $r_i - r_{i+1} = r_{i+2} < \dfrac{r_i}{2}$

Descending sequence:

$$\begin{array}{l} a \quad\;\; b \\ r_0 > r_1 > \quad \cdots \quad > r_i > r_{i+1} > r_{i+2} > \quad \cdots \quad > r_n > 0 \end{array}$$

$$r_i \bmod r_{i+1} \;=\; r_{i+2}$$

Property: $\frac{r_i}{2} > r_{i+2}$ $\Rightarrow$ $n \leq 2\log a$

Euclidian Algorithm

```
gcd($a,b$) {
    $x \leftarrow a$
    $y \leftarrow b$
    while ($y \neq 0$) {
        $r \leftarrow x \bmod y$
        $x \leftarrow y$
        $y \leftarrow r$
    }
    return $x$
}
```

Time complexity: $O(\log a)$ divisions

Relatively prime numbers

If $\gcd(a,b)=1$ then a,b are relatively prime

a and b have no common factors in their prime factorization

Example: 21, 22 are relatively prime

$$\gcd(21,22)=1$$

$21=3\cdot 7$ $\quad$ $22=2\cdot 11$

Least common multiple

$\text{lcm}(a,b)=$ smallest positive integer d such that $a\mid d$ and $b\mid d$

$a,b\in Z^{+}$

Examples: $\text{lcm}(3,4)=12$

$\text{lcm}(5,10)=10$

Applications of Number Theory

Linear combination:

if $a, b \in Z^+$ then there are $s, t \in Z$ such that

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

Example: $\gcd(6,14) = 2 = (-2)\cdot 6 + 1\cdot 14$

The linear combination can be found
by reversing the Euclidian algorithm steps

$$\gcd(252,198) = 18 = 4\cdot 252 - 5\cdot 198$$

$$\begin{aligned}
252 &= 1\cdot 198 + 54 \\
198 &= 3\cdot 54 + 36 \\
54 &= 1\cdot 36 + 18 \\
36 &= 2\cdot 18 + 0
\end{aligned}$$

$$\begin{aligned}
&\gcd(252,198) = 18 \\
&= 54 - 1\cdot 36 = 54 - 1\cdot(198 - 3\cdot 54) \\
&= 4\cdot 54 - 1\cdot 198 = 4\cdot(252 - 1\cdot 198) - 1\cdot 198 \\
&= 4\cdot 252 - 5\cdot 198
\end{aligned}$$

Linear congruences

We want to solve the equation for x

$$a \cdot x \equiv b (\bmod m)$$

$$\Downarrow$$

$$x \equiv ?\ (\bmod m)$$

Inverse of a: $\quad \bar{a}a \equiv 1 (\bmod m)$

$$\left.\begin{array}{r} a \cdot x \equiv b (\bmod m) \\ \bar{a} \equiv \bar{a} \bmod m \end{array}\right\} \Rightarrow \quad \bar{a}a \cdot x \equiv \bar{a}b (\bmod m)$$

$$\left.\begin{array}{r} \bar{a}a \equiv 1 (\bmod m) \\ x \equiv x (\bmod m) \end{array}\right\} \Rightarrow \bar{a}a \cdot x \equiv 1 \cdot x (\bmod m)$$

$$\Downarrow$$

$$x \equiv \bar{a}b (\bmod m)$$

Theorem: If a and m are relatively prime then the inverse $\overline{a}$ modulo m exists

Proof: $\gcd(a,m)=1=sa+tm$

$$\Downarrow$$

$$sa \equiv 1 (\bmod m)$$

$$\Downarrow$$

$$\overline{a} = s$$

End of proof

Example: solve equation $3x \equiv 4 (\bmod 7)$

$$a=3,\ b=4,\ m=7$$

Inverse of 3: $\overline{a}=-2$

$$\gcd(3,7)=1=\boxed{-2}\cdot 3+1\cdot 7 \Rightarrow -2\cdot 3\equiv 1 (\bmod m)$$

$$x \equiv \overline{a}b (\bmod m)$$

$$x \equiv -2\cdot 4 (\bmod 7) \equiv -8 (\bmod 7) \equiv 6 \bmod 7$$

Chinese remainder problem

$m_1, m_2, \ldots, m_n$:pairwise relatively prime

$$x \equiv a_1 \pmod{m_1}$$
$$x \equiv a_2 \pmod{m_2}$$
$$\vdots$$
$$x \equiv a_n \pmod{m_n}$$

Has unique solution for x modulo $m = m_1 \cdot m_2 \cdots m_n$

$$x \pmod{m}$$

Unique solution modulo $m = m_1 \cdot m_2 \cdots m_n$:

$$x = a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots + a_n M_n y_n$$

$$M_k = \frac{m}{m_k}$$

y_k :inverse of M_k modulo m_k

Explanation: y_k : inverse of M_k modulo m_k

$$M_k = \frac{m}{m_k} \qquad M_k y_k \equiv 1 \bmod m_k$$

$$x = a_1M_1y_1 + \overset{0(\bmod m_1)}{a_2M_2y_2} + \cdots + \overset{0(\bmod m_1)}{a_nM_ny_n}$$

$$x \equiv a_1M_1y_1 (\bmod m_1) \qquad M_{k\neq 1} \equiv 0(\bmod m_1)$$

$$x \equiv a_1 (\bmod m_1)$$

Similar for any m_j

Example:

$$x \equiv 2(\bmod 3)$$
$$x \equiv 3(\bmod 5)$$
$$x \equiv 2(\bmod 7)$$

$m = 3 \cdot 5 \cdot 7 = 105$

	$M_1 = m/3 = 105/3 = 35$	$y_1 = 2$
	$M_2 = m/5 = 105/5 = 21$	$y_2 = 1$
	$M_3 = m/7 = 105/7 = 15$	$y_3 = 1$

$$x = a_1M_1y_1 + a_2M_2y_2 + a_3M_3y_3$$
$$= 2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1$$
$$= 233 \equiv 23(\bmod 3 \cdot 5 \cdot 7) \equiv 23(\bmod 105)$$

An Application of Chinese remainder problem

Perform arithmetic with large numbers
using arithmetic modulo small numbers

Example: relatively prime numbers

$$m_1 = 99, \quad m_2 = 98, \quad m_3 = 97, \quad m_4 = 95$$

$$m = 99 \cdot 98 \cdot 97 \cdot 95 = 89{,}403{,}930$$

$$123{,}684 = (33, 8, 9, 89)$$

$123{,}684 \bmod 99 = 33$

$123{,}684 \bmod 98 = 8$

$123{,}684 \bmod 97 = 9$

$123{,}684 \bmod 95 = 89$

Any number smaller
than m has unique
representation

$$123{,}684 = (33, 8, 9, 89)$$

$$+ \quad + \quad + \quad +$$

$$+ \quad 413{,}456 = (32, 92, 42, 16)$$

$$(65 \bmod 99, 100 \bmod 98, 51 \bmod 97, 105 \bmod 95)$$

$$537{,}140 = (65, 2, 51, 10)$$

We obtain this by using the
Chinese remainder problem solution

Fermat's little theorem:

For any prime p and integer a
not divisible by p ($\gcd(a,p)=1$):

$$a^{p-1} \equiv 1 (\bmod p)$$

Example: $2^{340} \equiv 1(\text{mod})341$

$$a = 2 \quad p = 341$$

Proof:

Property 1:

p does not divide any of:

$$1a, \quad 2a, \quad 3a, \quad \ldots, \quad (p-1)a$$

Explanation:

Suppose p divides ka , $1 \le k \le p-1$

$$\Downarrow$$

$$\exists s \in Z: \quad ka = sp$$

$\gcd(a, p) = 1$
$1 \le k \le p-1$

Does not have p as prime factor

has p as prime factor

Contradicts fundamental theorem of arithmetic

Property 2:

any pair below is not congruent modulo p:

$$1a, \quad 2a, \quad 3a, \quad \ldots, \quad (p-1)a$$

Explanation:

Suppose $xa \equiv ya \pmod{p}$, $\quad 1 \le x < y \le p-1$

$$\Downarrow$$

$$\exists s \in Z: \quad ya = xa + sp$$

$$\Downarrow$$

$$(y-x)a = sp$$

$$\Downarrow$$

p divides $(y-x)a \qquad 1 \le y-x \le p-1$

Contradicts Property 1

Property 3:

$$1a \cdot 2a \cdot 3a \cdots (p-1)a \equiv 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod{p}$$

Explanation:

From Property 2

$$1a \equiv x_1 \pmod{p}, \quad 1 \le x_1 \le p-1$$

$$2a \equiv x_2 \pmod{p}, \quad 1 \le x_2 \le p-1$$

$$\vdots$$

$$(p-1)a \equiv x_{p-1} \pmod{p}, \quad 1 \le x_{p-1} \le p-1$$

$$x_i \ne x_j \qquad 1 \le i < j \le p-1$$

$$\Downarrow$$

$$x_1 \cdot x_2 \cdot x_3 \cdots x_{p-1} = 1 \cdot 2 \cdot 3 \cdots (p-1)$$

$$\Downarrow$$

$$1a \cdot 2a \cdot 3a \cdots (p-1)a \ \equiv \ 1 \cdot 2 \cdot 3 \cdots (p-1) \pmod{p}$$

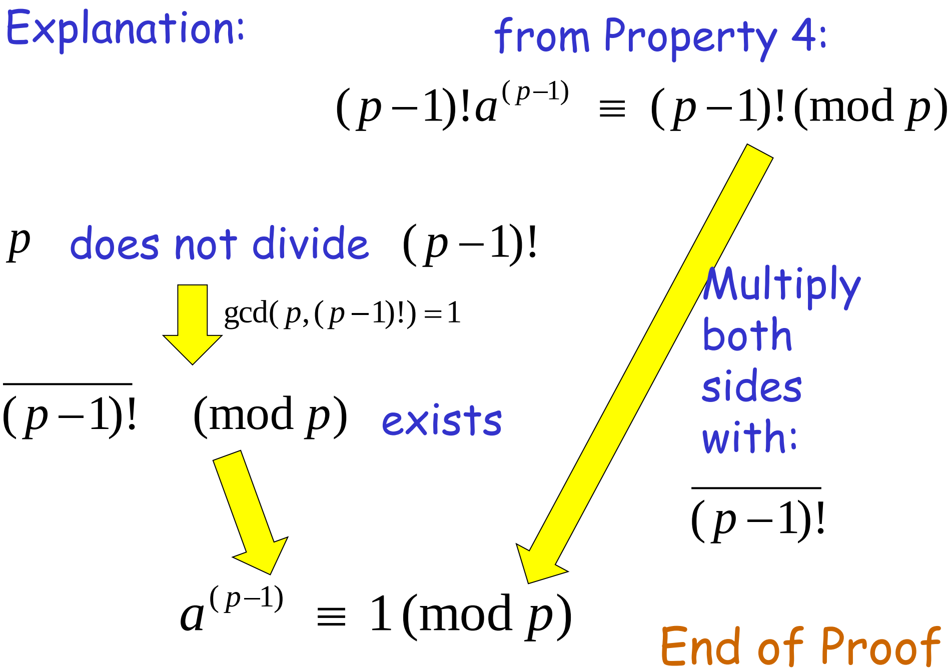
Property 4:

$$(p-1)!a^{(p-1)} \ \equiv \ (p-1)! \pmod{p}$$

(follows directly from property 3)

Property 5:

$$a^{(p-1)} \equiv 1 \pmod{p}$$



Explanation:

from Property 4:

$$(p-1)!\,a^{(p-1)} \equiv (p-1)! \pmod{p}$$

p does not divide $(p-1)!$

$\gcd(p,(p-1)!)=1$

$\overline{(p-1)!} \pmod{p}$ exists

Multiply both sides with: $\overline{(p-1)!}$

$$a^{(p-1)} \equiv 1 \pmod{p}$$

End of Proof

RSA (Rivest-Shamir-Adleman) cryptosystem

"MEET YOU IN THE PARK"

encryption

$f(x) = x^e \bmod n$

decryption

$f^{-1}(x) = x^d \bmod n$

"9383772909383637467"

$n = p \cdot q$

Large primes

n, e are public keys

p, q, d are private keys

Encryption example: $p = 43$ $q = 59$ $e = 13$

$$n = p \cdot q = 2537$$

$$\gcd(e, (p-1)(q-1)) = \gcd(13, 42 \cdot 58) = 1$$

Message to encrypt: "STOP"

Translate to equivalent numbers

"18 19 14 15"

Group into blocks of two numbers

"1819 1415"

Apply encryption function
to each block

“1819 1415”

$\downarrow$

$$f(x) = x^e \bmod n$$

$$= x^{13} \bmod 2537$$

Encrypted message: “2081 2182”

$$f(1819) = 1819^{13} \bmod 2537 = 2081$$

$$f(1415) = 1415^{13} \bmod 2537 = 2182$$

Message decryption

M :an original block of the message

“1819 1415”

$\downarrow$

“2081 2182”

C :respective encrypted block

$$C \equiv M^e (\bmod n)$$

We want to find M by knowing C, p, q, e

d :inverse of e modulo $(p-1)(q-1)$

$$de \equiv 1 (\bmod (p-1)(q-1))$$

$\Downarrow$ by definition of congruent

$$de = 1 + k(p-1)(q-1)$$

Inverse exists because $\gcd(e,(p-1)(q-1)) = 1$

$$\gcd(e,(p-1)(q-1)) = 1 = se + t(p-1)(q-1) \equiv se \bmod (p-1)(q-1)$$

$$\Downarrow$$

$$d = s$$

$$C \equiv M^e (\bmod n)$$

$$\Downarrow$$

$$C^d \equiv \left(M^e\right)^d (\bmod n)$$

$\Downarrow$ $de = 1 + k(p-1)(q-1)$

$$C^d \equiv M^{de} \equiv M^{1+k(p-1)(q-1)} (\bmod n)$$

Very likely it holds $\gcd(M, p) = 1$

(because p is a large prime and M is small)

$$\begin{tikzcd}
\gcd(M, p) = 1 \arrow[d, "\text{By Fermat's little theorem}"] \\
M^{p-1} \equiv 1 \pmod{p}
\end{tikzcd}$$

$$\begin{tikzcd}
M^{p-1} \equiv 1 \pmod{p} \arrow[d] \\
\left(M^{p-1}\right)^{k(q-1)} \equiv 1^{k(q-1)} \equiv 1 \pmod{p} \arrow[d, "M \equiv M \pmod{p}"] \\
M \cdot \left(M^{p-1}\right)^{k(q-1)} \equiv M \cdot 1 \pmod{p} \arrow[d] \\
M^{1+k(p-1)(q-1)} \equiv M \pmod{p}
\end{tikzcd}$$

We showed:

$$M^{1+k(p-1)(q-1)} \equiv M \pmod{p}$$

By symmetry, when replacing p with q:

$$M^{1+k(p-1)(q-1)} \equiv M \pmod{q}$$

By the Chinese remainder problem:

$$M^{1+k(p-1)(q-1)} \equiv M \pmod{pq} \equiv M \pmod{n}$$

We showed:

$$\left.\begin{array}{l} C^{d} \equiv M^{1+k(p-1)(q-1)} \pmod{n} \\ \\ M^{1+k(p-1)(q-1)} \equiv M \pmod{n} \end{array}\right\} \Rightarrow C^{d} \equiv M \pmod{n}$$

In other words:

$$M = C^{d} \bmod n$$

Decryption example: $p = 43 \qquad q = 59 \qquad e = 13$

$$n = p \cdot q = 2537$$

$$\gcd(e,(p-1)(q-1)) = \gcd(13, 42 \cdot 58) = 1$$

We can compute: $d = 937$

"2081 2182" $\qquad f^{-1}(C) = C^d \bmod n$

$2081^{937} \bmod 2537 = 1819 \quad \downarrow \quad \downarrow \quad 2182^{937} \bmod 2537 = 1415$

"1819 1415"

"18 19 14 15" = "STOP"