

Elementi di Teoria dei Numeri

Corso di Sicurezza dei Dati
a.a. 2020-21



Alfredo De Santis

Dipartimento di Informatica
Università di Salerno

ads@unisa.it

<http://www.di-srv.unisa.it/~ads>

Novembre 2020

Teoria dei numeri



Concetti preliminari per Crittografia a Chiave
Pubblica

- Aritmetica modulare
- Algoritmo di Euclide per il calcolo del gcd
- Calcolo dell'inversa moltiplicativa mod n
- Elevazione a potenza modulare
- Generazione di numeri primi
- Test di primalità

Teorema della divisione

Dati $a \in \mathbb{Z}$, $n \in \mathbb{N}$ esiste un'unica coppia (q, r)

con $0 \leq r \leq n-1$ tale che

- r è indicato con $a \bmod n$
 - $a=11, n=7, 11 = 1 \cdot 7 + 4 \rightarrow r = 11 \bmod 7 = 4$
 - $a=-11, n=7, -11 = (-2) \cdot 7 + 3 \rightarrow r = -11 \bmod 7 = 3$

Congruenze mod n

- a e b sono congruenti mod n ($a \equiv b \pmod{n}$) se
 $a \pmod{n} = b \pmod{n}$
- $73 \equiv 4 \pmod{23}$
 - $73 \pmod{23} = 4 \pmod{23}$
- $21 \equiv -9 \pmod{10}$
 - $21 \pmod{10} = -9 \pmod{10}$
- Se $a \equiv b \pmod{n}$ allora $n|(b-a)$
- Se $a \equiv b \pmod{n}$ allora $b \equiv a \pmod{n}$

L'insieme Z_n

- $[a]_n = \{a + kn, k \in \mathbb{Z}\}$
 - Insieme dei numeri che divisi per n danno lo stesso resto a mod n
 - Rappresentata dal più piccolo intero positivo che è in essa
 - $b \in [a]_n \longleftrightarrow b = a + kn \longleftrightarrow b - a = kn \longleftrightarrow n \mid (b - a) \longleftrightarrow b \equiv a \text{ mod } n$
- $Z_n = \{[a]_n, 0 \leq a \leq n-1\}$ per semplicità $Z_n = \{0, \dots, n-1\}$
 - Esempio: $Z_4 = \{[0]_4, [1]_4, [2]_4, [3]_4\}$
 - $[0]_4 = \{\dots, -12, -8, -4, 0, 4, 8, 12, \dots\}$
 - $[1]_4 = \{\dots, -11, -7, -3, 1, 5, 9, 13, \dots\}$
 - $[2]_4 = \{\dots, -10, -6, -2, 2, 6, 10, 14, \dots\}$
 - $[3]_4 = \{\dots, -9, -5, -1, 3, 7, 11, 15, \dots\}$

Aritmetica Modulare

➤ Addizione modulo n

$$\text{➤ } (a \bmod n) + (b \bmod n) = (a+b) \bmod n$$

➤ Esempio mod 8

+ 0 1 2 3 4 5 6 7	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

Aritmetica Modulare

➤ Addizione modulo n

$$\text{➤ } (a \bmod n) + (b \bmod n) = (a+b) \bmod n$$

➤ Proprietà

- Commutativa
- Associativa
- Identità
- Esistenza inversa additiva

(l'inversa additiva di x è y tale che $x+y \equiv 0 \pmod{n}$)

$(\mathbb{Z}_n, +)$ gruppo additivo mod n

Massimo Comune Divisore (gcd)

- d è il massimo comune divisore di a e n se
 - d è un divisore di a e n
 - ogni divisore di a e n è un divisore di d

Massimo Comune Divisore (gcd)

- d è il **massimo comune divisore** di a e n se
 - d è un divisore di a e n
 - ogni divisore di a e n è un divisore di d
- il **massimo comune divisore** d di a e n è il più piccolo intero positivo che può essere scritto nella forma

$$d = a \cdot x + n \cdot y$$

Massimo Comune Divisore (gcd)

- d è il **massimo comune divisore** di a e n se
 - d è un divisore di a e n
 - ogni divisore di a e n è un divisore di d
- il **massimo comune divisore** d di a e n è il più piccolo intero positivo che può essere scritto nella forma

$$d = a \cdot x + n \cdot y$$

- Proprietà
 - $\gcd(a, n) = \gcd(a, -n) = \gcd(-a, -n) = \gcd(|a|, |n|)$
 - $\gcd(a, 0) = |a|$
 - se $\gcd(a, n)=1$, a e n sono **relativamente primi**

L'insieme \mathbb{Z}_n^*

$$\mathbb{Z}_n^* = \{ [a]_n, 0 < a \leq n-1 \text{ e } \gcd(a,n)=1 \}$$

- Esempio: $\mathbb{Z}_4^* = \{ [1]_4, [3]_4 \}$
 - $[1]_4 = \{ \dots, -11, -7, -3, \textcolor{red}{1}, 5, 9, 13, \dots \}$
 - $[3]_4 = \{ \dots, -9, -5, -1, \textcolor{red}{3}, 7, 11, 15, \dots \}$
- Esempio: $\mathbb{Z}_8^* = \{ [1]_8, [3]_8, [5]_8, [7]_8 \}$
 - $[1]_8 = \{ \dots, -23, -15, -7, \textcolor{red}{1}, 9, 17, 25, \dots \}$
 - $[3]_8 = \{ \dots, -21, -13, -5, \textcolor{red}{3}, 11, 19, 30, \dots \}$

Aritmetica Modulare

➤ Moltiplicazione modulo n

$$\Rightarrow (a \bmod n) \cdot (b \bmod n) = (a \cdot b) \bmod n$$

➤ Proprietà

- Commutativa
- Associativa
- Distributiva
- Identità
- Esistenza inversa moltiplicativa

L'inversa moltiplicativa di x è y tale che $x \cdot y \equiv 1 \pmod{n}$

(\mathbb{Z}_n^*, \cdot) gruppo moltiplicativo mod n

Aritmetica mod 8

	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	0
2	2	3	4	5	6	7	0	1
3	3	4	5	6	7	0	1	2
4	4	5	6	7	0	1	2	3
5	5	6	7	0	1	2	3	4
6	6	7	0	1	2	3	4	5
7	7	0	1	2	3	4	5	6

Addizione

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

Moltiplicazione

Non tutti gli interi in \mathbb{Z}_8 hanno inversa moltiplicativa,
ma quelli in $\mathbb{Z}_8^* = \{ [1]_8, [3]_8, [5]_8, [7]_8 \}$ ce l'hanno

Aritmetica mod 7

Gli interi in Z_7 che hanno inversa moltiplicativa, sono quelli in

$$Z_7^* = \{ [1]_7, [2]_7, [3]_7, [4]_7, [5]_7, [6]_7 \}$$

	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



Algoritmo di Euclide

Descritto negli *Elementi* di Euclide (circa 300 A. C.)

Serve a calcolare il Massimo Comun Divisore

$$\gcd(30, 21) = ?$$

$$\gcd(63, 30) = ?$$

$$\gcd(4864, 3458) = ?$$



L'edizione del 1570



Algoritmo di Euclide

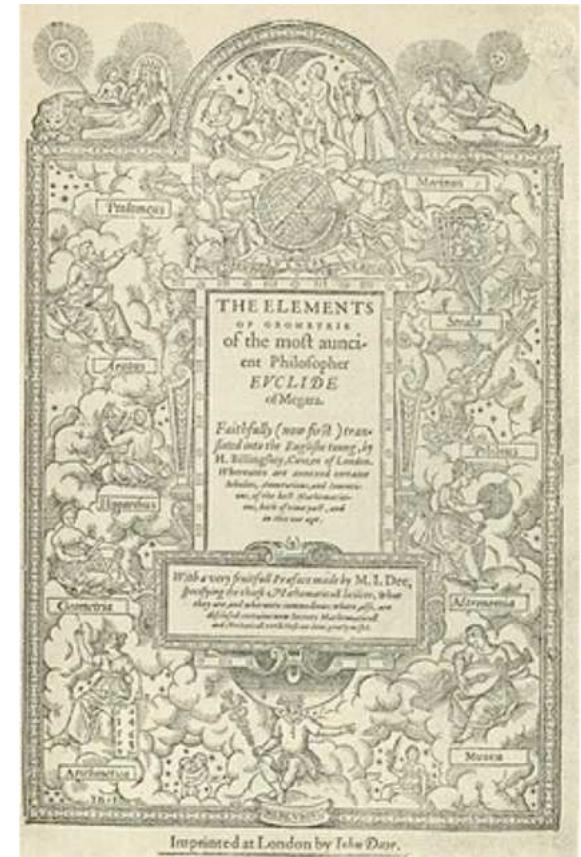
Descritto negli *Elementi* di Euclide (circa 300 A. C.)

Serve a calcolare il Massimo Comun Divisore

$$\gcd(30, 21) = 3$$

$$\gcd(63, 30) = 3$$

$$\gcd(4864, 3458) = 38$$



L'edizione 1570

Algoritmo di Euclide

Teorema della ricorsione del gcd

Per tutti gli interi $a \geq 0$ e $n > 0$

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

Algoritmo di Euclide

Teorema della ricorsione del gcd

Per tutti gli interi $a \geq 0$ e $n > 0$

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

Euclide (a,n)

if $n = 0$ then return a

else return **Euclide** ($n, a \bmod n$)

Algoritmo di Euclide: Esempi

Euclide (30, 21) = **Euclide** (21, 9)
= **Euclide** (9, 3)
= **Euclide** (3, 0) = 3

Euclide (4864, 3458) = **Euclide** (3458, 1406)
= **Euclide** (1406, 646)
= **Euclide** (646, 114)
= **Euclide** (114, 76)
= **Euclide** (76, 38)
= **Euclide** (38, 0) = 38

Algoritmo di Euclide: complessità

- Assumiamo $a > n \geq 0$
 - Se $a < n$, **Euclide** (a, n) chiama **Euclide** (n, a) e procede
 - Se $a = n$, **Euclide** (a, n) termina subito perché $a \bmod n = 0$
- Al massimo $\log n$ chiamate
 - Analisi complessa basata sui numeri di Fibonacci
- Per ogni chiamata $O((\log a)^2)$ operazioni su bit
- Totale: al massimo $O((\log a)^3)$ operazioni su bit
- **Euclide** (a, n) richiede al massimo $O((\log a)^2)$ operazioni su bit

Algoritmo di Euclide Esteso

Euclide-esteso (a,n)

```
if n = 0 then return (a, 1, 0)
(d', x', y') ← Euclide-esteso (n, a mod n)
(d, x, y) ← (d', y', x' - ⌊a/n⌋ y')
return (d, x, y)
```

- Oltre a computare $d = \gcd(a,n)$ computa anche due interi x, y tali che $d = \gcd(a,n) = a \cdot x + n \cdot y$
- Stesso running time asintotico di **Euclide** (a,n)

Algoritmo di Euclide Esteso

Euclide-esteso (a,n)

if $n = 0$ then return $(a, 1, 0)$

$(d', x', y') \leftarrow \text{Euclide-esteso} (n, a \bmod n)$

$(d, x, y) \leftarrow (d', y', x' - \lfloor a/n \rfloor y')$

return (d, x, y)

$$\begin{aligned} 3 &= \gcd(99, 78) \\ &= -11 \cdot 99 + 14 \cdot 78 \end{aligned}$$

a	n	d	x	y
99	78	3	-11	14
78	21	3	3	-11
21	15	3	-2	3
15	6	3	1	-2
6	3	3	0	1
3	0	3	1	0

Esempio: calcolo di $5^{-1} \bmod 7$

Euclide-esteso (a,n)

if $n = 0$ then return $(a, 1, 0)$

$(d', x', y') \leftarrow \text{Euclide-esteso} (n, a \bmod n)$

$(d, x, y) \leftarrow (d', y', x' - \lfloor a/n \rfloor y')$

return (d, x, y)

$$d = x \cdot a + y \cdot n$$

$$1 = -2 \cdot 7 + 3 \cdot 5$$

$$3 = 5^{-1} \bmod 7$$

a	n	d	x	y
7	5	1	-2	3
5	2	1	1	-2
2	1	1	0	1
1	0	1	1	0

Esempio: calcolo di $5^{-1} \bmod 11$

Euclide-esteso (a,n)

if $n = 0$ then return $(a, 1, 0)$

$(d', x', y') \leftarrow \text{Euclide-esteso} (n, a \bmod n)$

$(d, x, y) \leftarrow (d', y', x' - \lfloor a/n \rfloor y')$

return (d, x, y)

$$d = x \cdot a + y \cdot n$$

$$1 = 1 \cdot 11 + (-2) \cdot 5$$

$$-2 \equiv 5^{-1} \bmod 11$$

a	n	d	x	y
11	5	1	1	-2
5	1	1	0	1
1	0	1	1	0

$$-2 = 9 \bmod 11$$

$$9 = 5^{-1} \bmod 11$$

Soluzione di $ax \equiv 1 \pmod{8}$

Ha soluzioni se e solo se $\gcd(a, 8) = 1$

$$1 = 1^{-1} \pmod{8}$$

$$3 = 3^{-1} \pmod{8}$$

$$5 = 5^{-1} \pmod{8}$$

$$7 = 7^{-1} \pmod{8}$$

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

Soluzione di $ax \equiv 1 \pmod{7}$

Ha soluzioni se e solo se $\gcd(a, 7) = 1$

$$1 = 1^{-1} \pmod{7}$$

$$4 = 2^{-1} \pmod{7}$$

$$5 = 3^{-1} \pmod{7}$$

$$2 = 4^{-1} \pmod{7}$$

$$3 = 5^{-1} \pmod{7}$$

$$6 = 6^{-1} \pmod{7}$$

	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

Soluzione di $ax \equiv 1 \pmod{n}$

- Ha soluzioni se e solo se $\gcd(a,n) = 1$
- Se $\gcd(a,n) = 1$ c'è una unica soluzione mod n

x'

dove $1 = a \cdot x' + n \cdot y$ (da **Euclide-esteso** (a, n))

- Tale soluzione viene denotata con $a^{-1} \pmod{n}$
(inversa moltiplicativa di a , mod n)

Soluzione di $ax \equiv b \pmod{n}$

Dati a, b, n calcolare x

Esempi:

$3x \equiv 7 \pmod{8}$ soluzione 5

$2x \equiv 4 \pmod{8}$ soluzione 2,6

$4x \equiv 2 \pmod{8}$ nessuna soluzione

	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7
2	0	2	4	6	0	2	4	6
3	0	3	6	1	4	7	2	5
4	0	4	0	4	0	4	0	4
5	0	5	2	7	4	1	6	3
6	0	6	4	2	0	6	4	2
7	0	7	6	5	4	3	2	1

Soluzione di $ax \equiv b \pmod{n}$

- Ha soluzioni se e solo se $g \mid b$ $g = \gcd(a, n)$
- Se $g \mid b$ ci sono esattamente g distinte soluzioni mod n :

$$x' \frac{b}{g} + i \frac{n}{g} \quad \text{per } i = 0, 1, \dots, g-1$$

dove $g = a \cdot x' + n \cdot y$ (da **Euclide-esteso** (a, n))

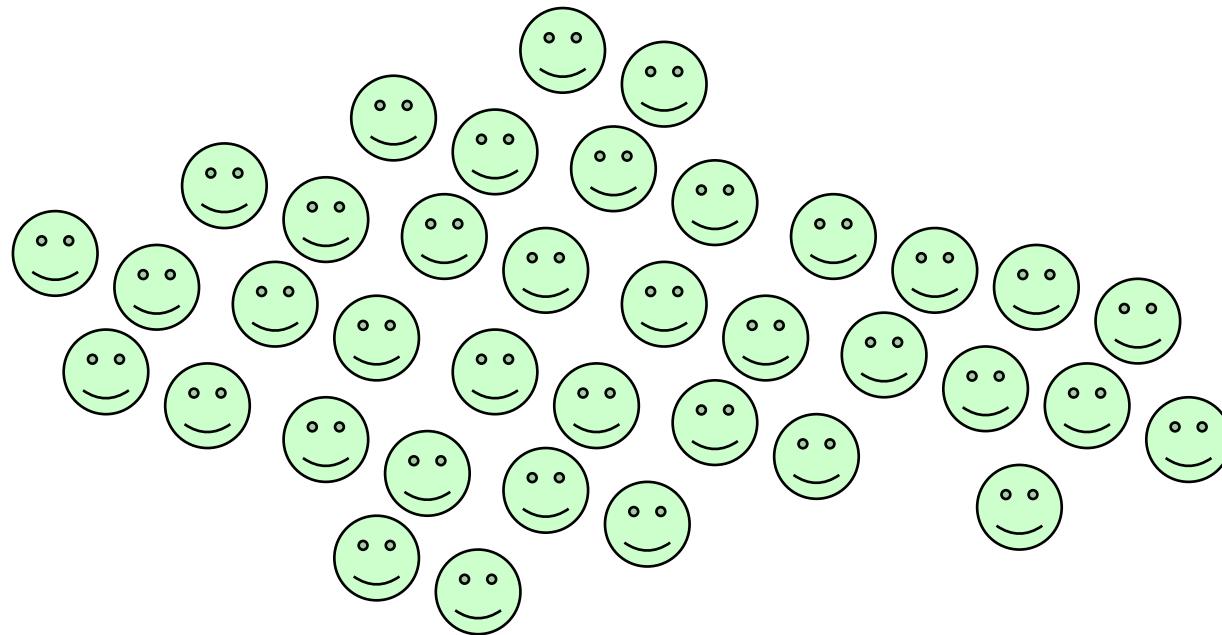
Teorema cinese del resto

III secolo d.C., matematico cinese Sun Zi

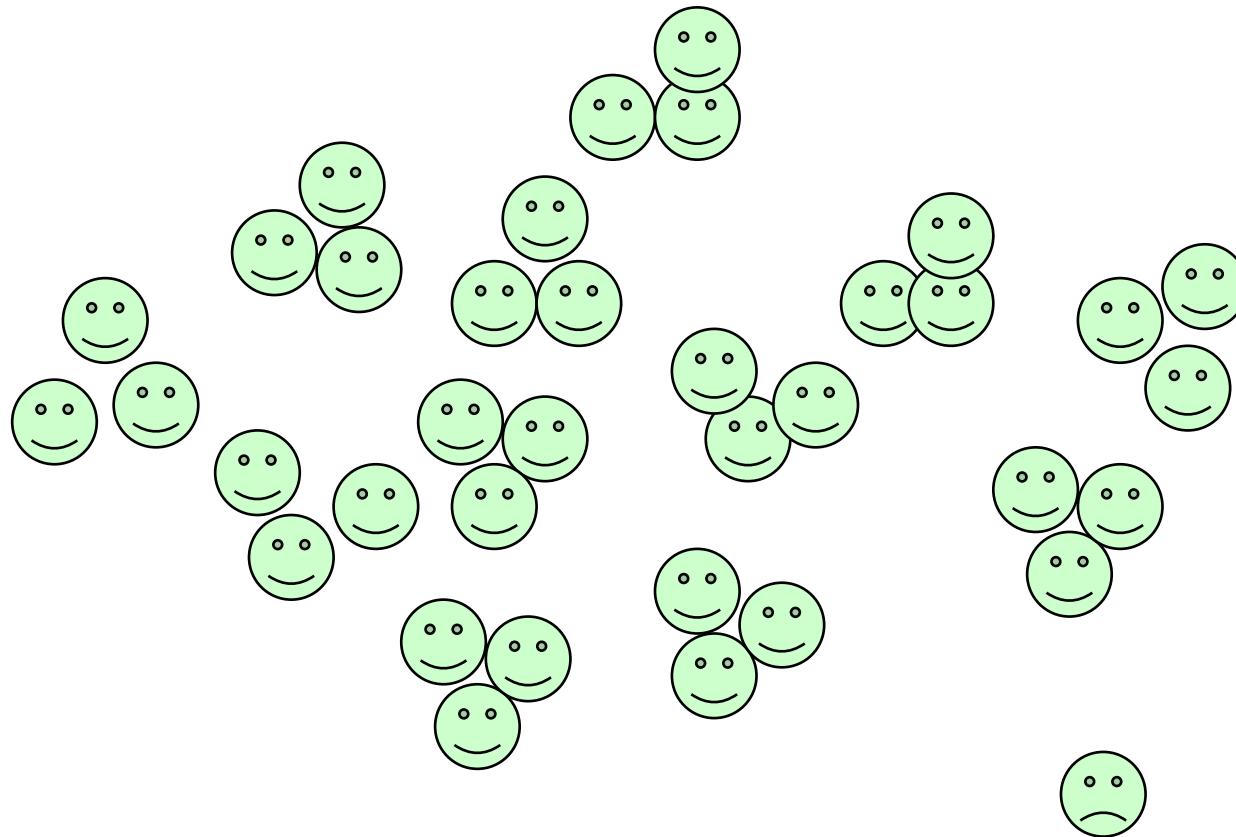
Imperatore cinese contava il suo esercito
mediante una sequenza di task

- Formare gruppi di 3. Riportare quanti non ci riuscivano.
- Formare gruppi di 5. Riportare quanti non ci riuscivano.
- Formare gruppi di 7. Riportare quanti non ci riuscivano.
- ...

Teorema cinese del resto

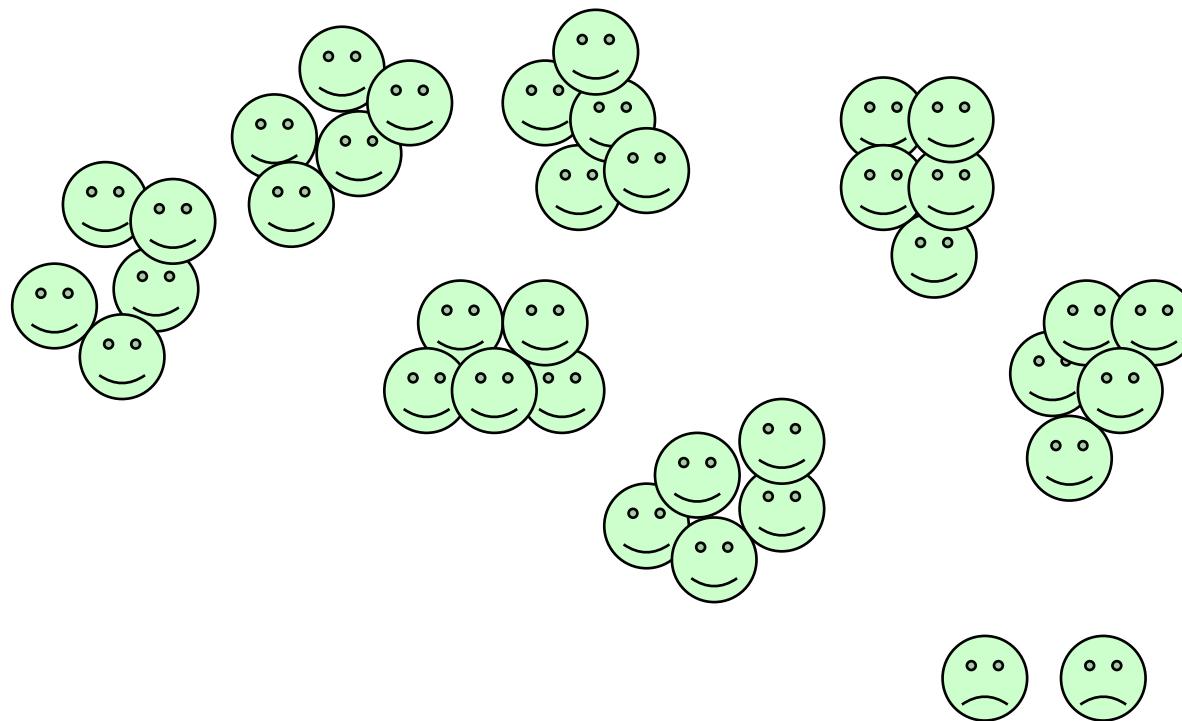


Teorema cinese del resto



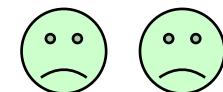
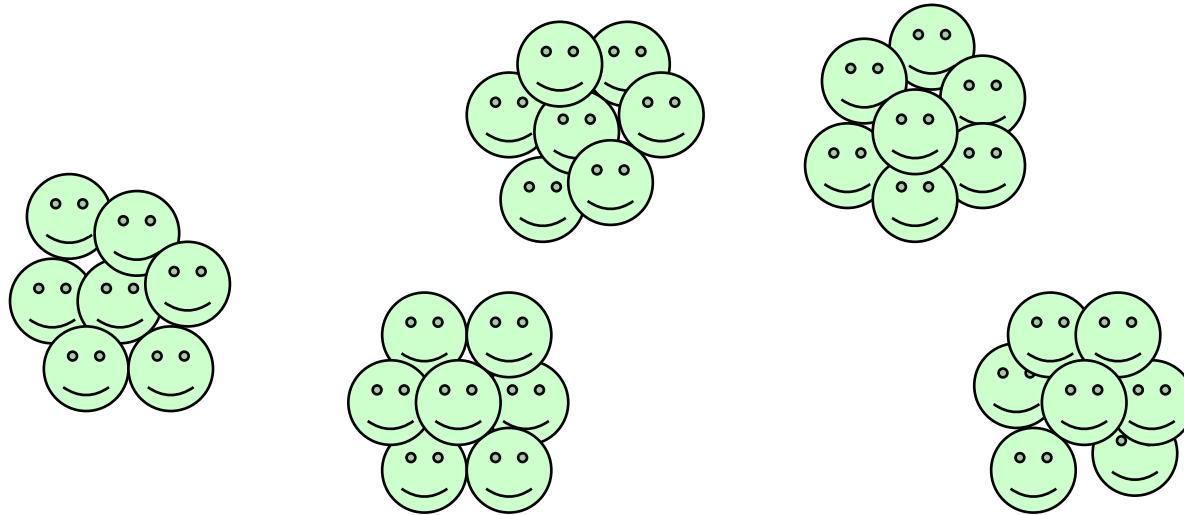
$$X \bmod 3 = 1$$

Teorema cinese del resto



$$X \bmod 5 = 2$$

Teorema cinese del resto



$$X \bmod 7 = 2$$

Teorema cinese del resto

Dati

- m_1, m_2, \dots, m_t interi positivi tali che $\gcd(m_i, m_j) = 1$, $i \neq j$
- $M = m_1 \cdot m_2 \cdots m_t$
- a_1, a_2, \dots, a_t interi

Esiste una sola soluzione modulo M al sistema di congruenze

$$\left\{ \begin{array}{l} X \equiv a_1 \pmod{m_1} \\ X \equiv a_2 \pmod{m_2} \\ \dots \\ X \equiv a_t \pmod{m_t} \end{array} \right.$$

$$X = \sum_{i=1}^t a_i \cdot M_i \cdot y_i \pmod{M}$$

$$M_i = M/m_i \quad y_i = M_i^{-1} \pmod{m_i}$$

Teorema cinese del resto

Esempio

$$\begin{cases} X \equiv 2 \pmod{5} \\ X \equiv 3 \pmod{13} \end{cases}$$

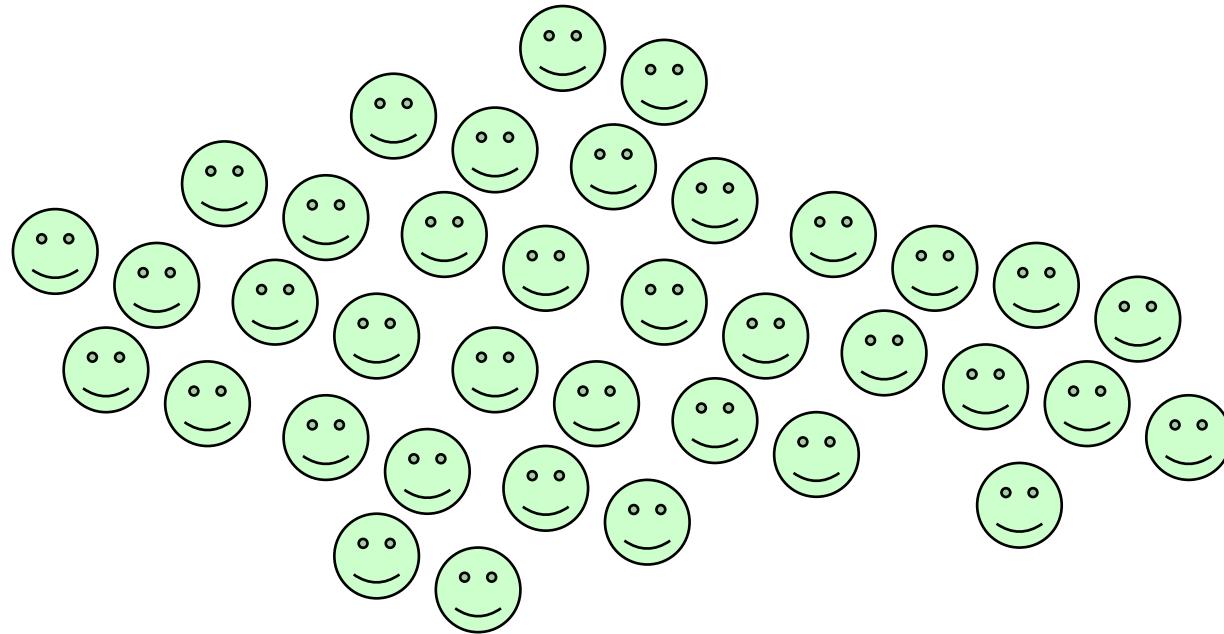
$$X = \sum_{i=1}^2 a_i \cdot M_i \cdot y_i \pmod{65}$$

$M_i = M/m_i \quad y_i = M_i^{-1} \pmod{m_i}$			
$a_1=2$	$m_1=5$	$M_1=13$	$y_1 = 13^{-1} \pmod{5} = 2$
$a_2=3$	$m_2=13$	$M_2=5$	$y_2 = 5^{-1} \pmod{13} = 8$

Soluzione del sistema:

$$\begin{aligned} X &= a_1 M_1 y_1 + a_2 M_2 y_2 \pmod{65} \\ &= 2 \cdot 3 \cdot 13 + 3 \cdot 5 \cdot 8 \pmod{65} \\ &= 42 \pmod{65} \end{aligned}$$

Teorema cinese del resto



Quanti sono?

Teorema cinese del resto

Esempio

$$\begin{cases} X \equiv 1 \pmod{3} \\ X \equiv 2 \pmod{5} \\ X \equiv 2 \pmod{7} \end{cases}$$

$$X = \sum_{i=1}^2 a_i \cdot M_i \cdot y_i \pmod{65}$$

$M_i = M/m_i \quad y_i = M_i^{-1} \pmod{m_i}$			
$a_1=1$	$m_1=3$	$M_1=35$	$y_1 = 35^{-1} \pmod{3} = 2$
$a_2=2$	$m_2=5$	$M_2=21$	$y_2 = 21^{-1} \pmod{5} = 1$
$a_3=2$	$m_3=7$	$M_3=15$	$y_3 = 15^{-1} \pmod{7} = 1$

Soluzione del sistema:

$$\begin{aligned} X &= a_1 M_1 y_1 + a_2 M_2 y_2 + a_3 M_3 y_3 \pmod{105} \\ &= 1 \cdot 35 \cdot 2 + 2 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1 \pmod{105} \\ &= 1 \cdot 70 + 2 \cdot 21 + 2 \cdot 15 \pmod{105} \\ &= 142 \pmod{105} = 37 \pmod{105} \end{aligned}$$

Teoria dei numeri



Concetti preliminari per Crittografia a Chiave Pubblica

- Aritmetica modulare
- Algoritmo di Euclide per il calcolo del gcd
- Calcolo dell'inversa moltiplicativa mod n
- **Elevazione a potenza modulare**
- Generazione di numeri primi
- Test di primalità

Elevazione a potenza modulare

Calcolo di $x^y \text{ mod } z$

- Metodo naive
- Metodo left-to-right
- Metodo right-to-left



Elevazione a potenza modulare

Metodo naive

Calcolo di $x^y \text{ mod } z$

Potenza_Modulare_naive (x, y, z)

```
a ← 1
for i = 1 to y do
    a ← (a · x) mod z
return a
```

Elevazione a potenza modulare

Metodo naive

Calcolo di $x^y \text{ mod } z$

Potenza_Modulare_naive (x, y, z)

```
a ← 1  
for i = 1 to y do  
    a ← (a · x) mod z  
return a
```



Se y è di 1024 bit, occorrono $\approx 2^{1024}$ operazioni

Esponenziale nella lunghezza dell'esponente

Elevazione a potenza modulare

Metodo left-to-right

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea: $y = y_0 + 2(y_1 + 2(y_2 + \dots + 2(y_{t-1} + 2y_t))))$

Esempio:

$$40 = 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5$$

$$40 = 0 + (2(0 + 2(0 + (2(1 + 2(0 + 2 \cdot 1))))))$$

Elevazione a potenza modulare

Metodo left-to-right

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea:

$$y = y_0 + 2(y_1 + 2(y_2 + \dots + 2(y_{t-1} + 2y_t))))$$

$$x^y = x^{y_0 + 2(y_1 + 2(y_2 + \dots + 2(y_{t-1} + 2y_t))))}$$

$$= x^{y_0} x^{2(y_1 + 2(y_2 + \dots + 2(y_{t-1} + 2y_t))))}$$

$$= x^{y_0} (x^{y_1 + 2(y_2 + \dots + 2(y_{t-1} + 2y_t)))))^2$$

$$= x^{y_0} (x^{y_1} (x^{y_2 + \dots + 2(y_{t-1} + 2y_t)}))^2)^2$$

$$= x^{y_0} (x^{y_1} (\dots (x^{y_{t-1}} (x^{y_t})^2)^2 \dots)^2$$

Elevazione a potenza modulare

Metodo left-to-right

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea: $y = y_0 + 2(y_1 + \dots + 2(y_{t-1} + 2y_t))) \dots)$

$$x^y = x^{y_0} (\dots (x^{y_{t-1}} (x^{y_t})^2)^2 \dots)^2$$

Esempio: x^{40}

$$40 = 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5$$

$$40 = 0 + (2(0 + 2(0 + (2(1 + 2(0 + 2 \cdot 1)))))))$$

$$x^{40} = x^0 (x^0 (x^0 (x^1 (x^0 (x^1)^2)^2)^2)^2)^2$$

Elevazione a potenza modulare

Metodo left-to-right

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea: $y = y_0 + 2(y_1 + \dots + 2(y_{t-1} + 2y_t))) \dots)$

$$x^y = x^{y_0} (\dots (x^{y_{t-1}} (x^{y_t})^2)^2 \dots)^2$$

Potenza_Modulare (x, y, z)

```
a ← 1
for i = t downto 0 do
    a ← (a · a) mod z
    if  $y_i = 1$  then a ← (a · x) mod z
return a
```



Se y è di 1024 bit, occorrono ≈ 1024 operazioni
Polinomiale nella lunghezza dell'esponente

Elevazione a potenza modulare

Metodo left-to-right

Potenza_Modulare (x, y, z)

```
a ← 1
for i = t downto 0 do
    a ← (a · a) mod z
    if  $y_i = 1$  then a ← (a · x) mod z
return a
```

$$3^{40} \bmod 73 = 8$$

$$y_5=1, y_4=0, y_3=1, y_2=0, y_1=0, y_0=0$$

Elevazione a potenza modulare

Metodo right-to-left

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea:

$$\begin{aligned}x^y &= x^{2^0 y_0 + 2^1 y_1 + \dots + 2^t y_t} \\&= x^{2^0 y_0} \cdot x^{2^1 y_1} \cdot \dots \cdot x^{2^t y_t} \\&= (x^{2^0})^{y_0} \cdot (x^{2^1})^{y_1} \cdot \dots \cdot (x^{2^t})^{y_t}\end{aligned}$$

Elevazione a potenza modulare

Metodo right-to-left

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea: $x^y = (x^{2^0})^{y_0} \cdot (x^{2^1})^{y_1} \cdot \dots \cdot (x^{2^t})^{y_t}$

Esempio: x^{40}

$$40 = 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5$$
$$x^{40} = (x^1)^0 \cdot (x^2)^0 \cdot (x^4)^0 \cdot (x^8)^1 \cdot (x^{16})^0 \cdot (x^{32})^1$$

Elevazione a potenza modulare

Metodo right-to-left

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea: $x^y = (x^{2^0})^{y_0} \cdot (x^{2^1})^{y_1} \cdot \dots \cdot (x^{2^t})^{y_t}$

Esempio: x^{40}

$$40 = 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5$$
$$x^{40} = (x^1)^0 \cdot (x^2)^0 \cdot (x^4)^0 \cdot (x^8)^1 \cdot (x^{16})^0 \cdot (x^{32})^1$$

$$x^1$$

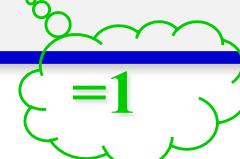
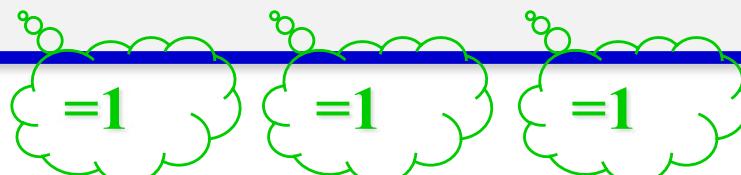
$$x^2$$

$$x^4$$

$$x^8$$

$$x^{16}$$

$$x^{32}$$



Elevazione a potenza modulare

Metodo right-to-left

Calcolo di $x^y \text{ mod } z$ $y = y_0 2^0 + y_1 2^1 + \dots + y_t 2^t$



Idea: $x^y = (x^{2^0})^{y_0} \cdot (x^{2^1})^{y_1} \cdot \dots \cdot (x^{2^t})^{y_t}$

Esempio: x^{40}

$$40 = 0 \cdot 2^0 + 0 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 + 0 \cdot 2^4 + 1 \cdot 2^5$$
$$x^{40} = (x^1)^0 \cdot (x^2)^0 \cdot (x^4)^0 \cdot (x^8)^1 \cdot (x^{16})^0 \cdot (x^{32})^1$$
$$x^{40} = \quad \quad \quad x^8 \quad \cdot \quad \quad \quad x^{32}$$

Elevazione a potenza modulare

Metodo right-to-left

Potenza_Modulare (x, y, z)

```
if y = 0 then return 1  
X ← x; P ← 1  
if y0 = 1 then P ← x  
for i = 1 to t do  
    X ← X · X mod z  
    if yi=1 then P ← P · X mod z  
return P
```

Polinomiale nella
lunghezza
dell'esponente

$5^{596} \text{mod } 1234$

i	0	1	2	3	4	5	6	7	8	9
y _i	0	0	1	0	1	0	1	0	0	1
X	5	25	625	681	1011	369	421	779	947	925
P	1	1	625	625	67	67	1059	1059	1059	1013

Elevazione a potenza modulo numero composto $N=pq$

Calcolo "più veloce" di $x^y \bmod N$

- Calcolo di $a = x^y \bmod p$
- Calcolo di $b = x^y \bmod q$
- Formula di Garner

$$x^y \bmod N = q \left(q^{-1} (a-b) \bmod p \right) + b$$

Teoria dei numeri



Concetti preliminari per Crittografia a Chiave Pubblica

- Aritmetica modulare
- Algoritmo di Euclide per il calcolo del gcd
- Calcolo dell'inversa moltiplicativa mod n
- Elevazione a potenza modulare
- Generazione di numeri primi
- Test di primalità

Numeri primi

Un intero $p > 1$ è un **numero primo** se e solo se i suoi unici divisori sono 1 e sé stesso

➤ Esempi: 2, 3, 5, 7, 11, 13, 17, 19, ...

Teorema fondamentale dell'aritmetica

Ogni intero composto $n > 1$ può essere scritto in modo unico come prodotto di potenze di primi

$$n = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$$

p_i primo, e_i intero positivo

Numeri primi

Un intero $p > 1$ è un **numero primo** se e solo se i suoi unici divisori sono 1 e sé stesso

➤ Esempi: 2, 3, 5, 7, 11, 13, 17, 19, ...

Teorema fondamentale dell'aritmetica

Ogni intero composto $n > 1$ può essere scritto in modo unico come $\dots \cdot p_{k-1}^{e_{k-1}} \cdot p_k^{e_k}$:

Prima dimostrazione:

Gauss, *Disquisitiones Arithmeticae*, 1798
(in latino, quando aveva 21 anni)

$$n = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$$

p_i primo, e_i intero positivo

Funzione di Eulero

$$\mathbb{Z}_n^* = \{ [a]_n, 0 < a \leq n-1 \text{ e } \gcd(a,n)=1 \}$$

$\phi(n)$ = cardinalità di \mathbb{Z}_n^* (funzione di Eulero)

- $\phi(p) = p-1$ se p primo
- $\phi(pq) = (p-1)(q-1)$ se p,q primi
- $\phi(n) = n \cdot \left(1 - \frac{1}{p_1}\right) \cdot \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_k}\right)$ se $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$,
 p_i primo, $e_i > 0$

Funzione di Eulero

$$\mathbb{Z}_n^* = \{ [a]_n, 0 < a \leq n-1 \text{ e } \gcd(a,n)=1 \}$$

$\phi(n)$ = cardinalità di \mathbb{Z}_n^* (funzione di Eulero)

➤ $\phi(p) = p-1$ se p primo

➤ $\phi(pq) = (p-1)(q-1)$ se p,q primi

➤ $\phi(n) = n \cdot \left(1 - \frac{1}{p_1}\right) \cdot \left(1 - \frac{1}{p_2}\right) \cdots \left(1 - \frac{1}{p_k}\right)$ se $n = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$,
 p_i primo, $e_i > 0$

$$\phi(p^2) = p(p-1)$$

$$\phi(p^3) = p^2(p-1)$$

se p primo

Funzione di Eulero

$$\phi(n) = \text{cardinalità di } Z_n^*$$

Esempi:

$$\phi(49) = 7 \cdot 6 = 42$$

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	32	33	34	35
36	37	38	39	40	41	42
43	44	45	46	47	48	49

$$\phi(35) = 6 \cdot 4 = 24$$

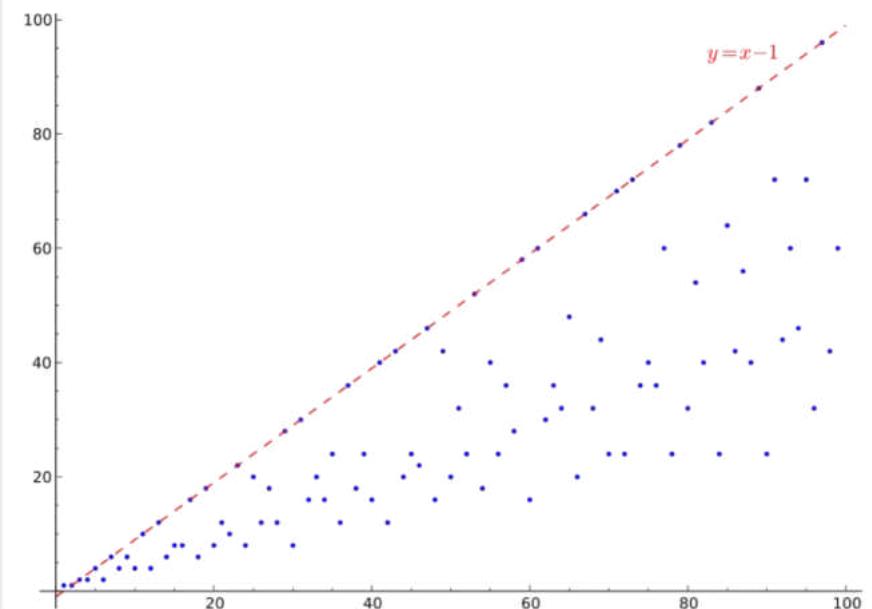
1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35

Funzione di Eulero

Valore di $\varphi(n)$ per $n = 1, 2, \dots, 100$

decine

(n)	0	1	2	3	4	5	6	7	8	9	unità
0	1	1	2	2	4	2	6	4	6		
10	4	10	4	12	6	8	8	16	6	18	
20	8	12	10	22	8	20	12	18	12	28	
30	8	30	16	20	16	24	12	36	18	24	
40	16	40	12	42	20	24	22	46	16	42	
50	20	32	24	52	18	40	24	36	28	58	
60	16	60	30	36	32	48	20	66	32	44	
70	24	70	24	72	36	40	36	60	24	78	
80	32	54	40	82	24	64	42	56	40	88	
90	24	72	44	60	46	72	32	96	42	60	



Teorema di Eulero

- Per ogni $a \in \mathbb{Z}_n^*$, $a^{\phi(n)} = 1 \pmod{n}$
- Esempi:
 - $a=3, n=10, \phi(10)=4 \rightarrow 3^4 = 81 = 1 \pmod{10}$
 - $a=2, n=11, \phi(11)=10 \rightarrow 2^{10} = 1024 = 1 \pmod{11}$

Teorema di Fermat

➤ Se p è primo, per ogni $a \in \mathbb{Z}_p^*$,

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



➤ Se p è primo, per ogni $a \in \mathbb{Z}_p$

$$a^p = a \pmod{p}$$

Esempi:

➤ $a=7, p=19 \rightarrow 7^{18} = 1 \pmod{19}$

➤ $a=10, p=5 \rightarrow 10^5 = 10 \pmod{5} = 0 \pmod{5}$

Potenze in \mathbb{Z}_{19}^*

Generazione di un primo 'grande'

1. Genera a caso un dispari p di grandezza appropriata
2. Testa se p è primo
3. Se p è composto, go to 1.

Generazione di un primo 'grande'

1. Genera a caso un dispari p di grandezza appropriata
2. Testa se p è primo
3. Se p è composto, go to 1.

Come testare se un numero p è primo?

Che probabilità abbiamo che p sia primo?

Distribuzione dei numeri primi

- $\pi(x)$ = numero di primi in $[2,x]$
- Teorema dei numeri primi: $\lim_{x \rightarrow \infty} \frac{\pi(x)}{x/\ln x} = 1$
 $\pi(x)$ è circa $x/\ln x$

Distribuzione dei numeri primi

- $\pi(x)$ = numero di primi in $[2,x]$
- Teorema dei numeri primi: $\lim_{x \rightarrow \infty} \frac{\pi(x)}{x/\ln x} = 1$
 $\pi(x)$ è circa $x/\ln x$
- Esempio: $\pi(10^{23}) = 1.925.320.391.606.803.968.923$

$$\frac{\pi(10^{23})}{10^{23}/\ln 10^{23}} \approx 1,020 \text{ (cioè 2% in meno)}$$

Distribuzione dei numeri primi

- $\pi(x)$ = numero di primi in $[2,x]$
- Teorema dei numeri primi: $\lim_{x \rightarrow \infty} \frac{\pi(x)}{x/\ln x} = 1$
 $\pi(x)$ è circa $x/\ln x$
- Esistono limitazioni, ad esempio, per $x > 355.990$

$$\frac{x}{\ln x} \left(1 + \frac{1}{\ln x}\right) < \pi(x) < \frac{x}{\ln x} \left(1 + \frac{1}{\ln x} + \frac{2,51}{(\ln x)^2}\right)$$

Scelta di un primo di 512 bit

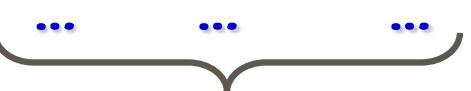
Scelta intero di 512 bit dispari

1  1

510 bit scelti a caso

Scelta di un primo di 512 bit

Scelta intero di 512 bit dispari

1  1

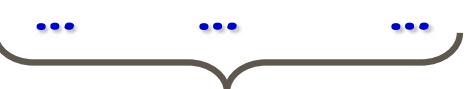
510 bit scelti a caso

Probabilità che sia primo

$$= \frac{\#primi \text{ in } [2^{511}, 2^{512}-1]}{\#\text{numeri in insieme}} \approx \frac{\frac{2^{512}}{\ln 2^{512}} - \frac{2^{511}}{\ln 2^{511}}}{2^{510}} \approx \frac{1}{177,79}$$

Scelta di un primo di 512 bit

Scelta intero di 512 bit dispari

1  1

510 bit scelti a caso

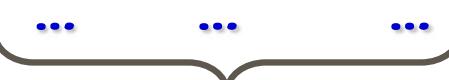
Probabilità che sia primo

$$= \frac{\#\text{primi in } [2^{511}, 2^{512}-1]}{\#\text{numeri in insieme}} \approx \frac{\frac{2^{512}}{\ln 2^{512}} - \frac{2^{511}}{\ln 2^{511}}}{2^{510}} \approx \frac{1}{177,79}$$

Numero medio di tentativi per trovare primo 177,79

Scelta di un primo di 1024 bit

Scelta intero di 1024 bit dispari

1  1

1022 bit scelti a caso

Probabilità che sia primo

$$= \frac{\# \text{primi in } [2^{1023}, 2^{1024}-1]}{\# \text{numeri in insieme}} \approx \frac{\frac{2^{1024}}{\ln 2^{1024}} - \frac{2^{1023}}{\ln 2^{1023}}}{2^{1022}} \approx \frac{1}{355,23}$$

Numero medio di tentativi per trovare primo 355,23

Esercizio

Scelta di un primo di 2048 bit

- Come si sceglie?
- Calcola probabilità di trovare primo
- Calcola numero medio tentativi

Teoria dei numeri



Concetti preliminari per Crittografia a Chiave Pubblica

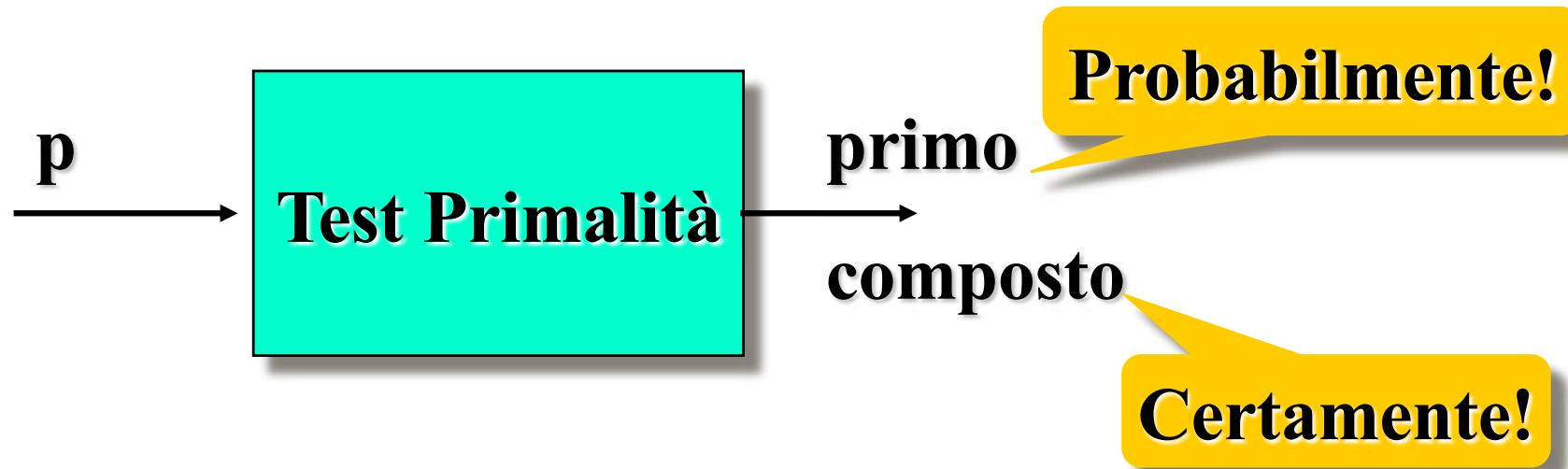
- Aritmetica modulare
- Algoritmo di Euclide per il calcolo del gcd
- Calcolo dell'inversa moltiplicativa mod n
- Elevazione a potenza modulare
- Generazione di numeri primi
- Test di primalità

Test di Primalità

Due tipi di test:

- Probabilistici 
- Deterministici

Test di primalità probabilistici



- Il test può sbagliare
 - numero composto viene dichiarato primo
 - probabilità di errore $\leq 1/2$
- Ripetendo il test indipendentemente t volte, probabilità di errore $\leq (1/2)^t$

Test di primalità probabilistici



Insieme di **witness** $W(p)$

- dato $a \in [1, p-1]$ è facile verificare se $a \in W(p)$
- se p è primo allora $W(p)$ è vuoto
- se p è composto allora $|W(p)| \geq p/2$





Test di primalità probabilistici

Test di Solovay-Strassen

- Probabilità di errore $\leq (1/2)^t$



➤ Robert M. Solovay e
Volker Strassen [1977]

Test di Miller-Rabin

- Il più usato in pratica
- Il più veloce
- Probabilità di errore $\leq (1/4)^t$



➤ Gary Miller, deterministico, basato
su Riemann hypothesis [1976]
➤ Michel Rabin, modificato in
probabilistico [1980]

Test di primalità deterministici

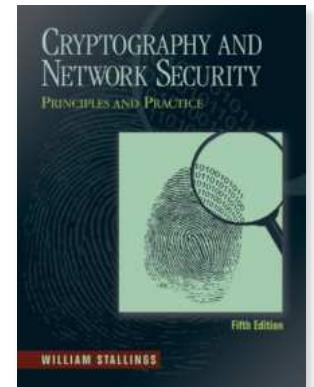
- Fino al 2002, non erano noti test deterministici efficienti (polinomiali) 
- Test deterministico AKS $O(\log^{12+\varepsilon} n)$

M. Agrawal, N. Kayal e N. Saxena,
"PRIMES is in P"


- Variante AKS con complessità $O(\log^{6+\varepsilon} n)$
- Miller-Rabin ancora usato perché più efficiente

Bibliografia

- **Cryptography and Network Security**
by W. Stallings, 2010
 - cap. 4 (Finite Fields)
 - cap. 8 (Introduction to Number Theory)
- **Introduction to Algorithms**
by Cormen, Leiserson, Rivest (I ed)
 - cap 31 (Number Theory)



Domande?

