

Un po' di teoria dei numeri

Applicazione alla crittografia RSA

Christian Ferrari

Liceo di Locarno

Matematica

Sommario

1 L'aritmetica modulare di \mathbb{Z}_n

- Le congruenze
- L'anello \mathbb{Z}_n
- Le potenze in \mathbb{Z}_n e algoritmo di Legendre
- MCD e algoritmi di Euclide ed Euclide esteso
- L'inverso in \mathbb{Z}_n
- La funzione $\varphi(n)$ di Euler
- Il piccolo teorema di Fermat e il teorema di Euler
- Un complemento sulle strutture algebriche

2 Le diverse basi numeriche

- La notazione posizionale e le diverse basi
- Il logaritmo e la lunghezza di un numero

3 I numeri primi

- Alcuni teoremi sui numeri primi
- Alcuni criteri di primalità
- La fattorizzazione dei numeri primi: un problema **NP**

Sommario

Gli aspetti della **teoria dei numeri** sviluppati precedentemente sono (in parte) utilizzata per implementare il seguente protocollo di crittografia:

⇒ Il protocollo di crittografia RSA

- La generazione delle chiavi
- Dal testo al messaggio numerico e viceversa
- Cifrare e decifrare
- Il teorema RSA
- La sicurezza di RSA
- La firma digitale



Potenze in \mathbb{Z}_n

Il calcolo delle potenze in \mathbb{Z}_n richiede di:

- calcolare a^r ($r \in \mathbb{N}^*$);
- calcolare il resto della divisione euclidea $a^r : n$.

Questo processo diventa computazionalmente complesso se i numeri in gioco sono grandi. Il seguente teorema (secondo punto) permette di ovviare al problema.

Teorema

Se $a_1 \equiv a_2 \pmod{n}$ e $b_1 \equiv b_2 \pmod{n}$, allora:

- $a_1 + b_1 \equiv a_2 + b_2 \pmod{n}$;
- $a_1 b_1 \equiv a_2 b_2 \pmod{n}$.

Potenze in \mathbb{Z}_n

Un **algoritmo** è un processo che, partendo da dei dati iniziali, viene ripetuto così da ottenere il risultato cercato. Ogni ripetizione è detta passo o **iterazione**.

Ecco un algoritmo semplice per il calcolo delle potenze che sfrutta il teorema precedente:

$$a^k = aa^{k-1} \text{ e } \begin{cases} a \equiv b_1 \pmod{n} & (b_1 = a) \\ a^{k-1} \equiv b_{k-1} \pmod{n} \end{cases} \implies a^k \equiv b_1 b_{k-1} \pmod{n}$$

Algoritmo semplice per le potenze in \mathbb{Z}_n

- START $k = 1$: porre $b_1 = a$;
- PASSO $k = 2$: $a^2 \equiv b_1 b_1 \pmod{n}$;
- PASSO $k = 3$: $a^3 \equiv b_1 b_2 \pmod{n}$;
- PASSO k generale: $a^k \equiv b_1 b_{k-1} \pmod{n}$;
- STOP : quando $k = n$.

Potenze in \mathbb{Z}_n

Un algoritmo più efficace, noto come **algoritmo di Legendre**, si basa sull'idea seguente per calcolare

$$a^r \equiv b \pmod{n}.$$

- Si scrive $r = \sum_{i=0}^d r_i 2^i$ dove $r_i \in \{0, 1\}$ (ossia si scompone r in base 2).
- Quindi $a^r = \prod_{i=0}^d a^{(r_i 2^i)}$.
- Si sfrutta poi che
 - se $a^2 \equiv b_2 \pmod{n}$ allora $a^4 \equiv b_2 b_2 \pmod{n} [= b_4 \pmod{n}]$;
 - e quindi $a^8 \equiv b_4 b_4 \pmod{n} [= b_8 \pmod{n}]$;
 - ... così da ricomporre $a^r \equiv b \pmod{n}$ come prodotto di potenze del tipo 2^i calcolate modulo n .

Potenze in \mathbb{Z}_n

Ecco un esempio: 7^{18} in \mathbb{Z}_{15} .

- Scriviamo 18 in base 2: $18 = 2^4 + 2^1 = 16 + 2$.
- Allora $7^{18} = 7^{16+2} = 7^{16} \cdot 7^2$.
- Calcoliamo

$$\begin{aligned} 7 &\equiv 7 \pmod{15} \\ 7^2 = 7 \cdot 7 &\equiv 7 \cdot 7 \pmod{15} \equiv 4 \pmod{15} \\ 7^4 = 7^2 \cdot 7^2 &\equiv 4 \cdot 4 \pmod{15} \equiv 1 \pmod{15} \\ 7^8 = 7^4 \cdot 7^4 &\equiv 1 \cdot 1 \pmod{15} \equiv 1 \pmod{15} \\ 7^{16} = 7^8 \cdot 7^8 &\equiv 1 \cdot 1 \pmod{15} \equiv 1 \pmod{15} \end{aligned}$$

- Quindi

$$7^{16} \cdot 7^2 \equiv 4 \cdot 1 \pmod{15} \implies 7^{18} \equiv 4 \pmod{15}$$

e in \mathbb{Z}_{15} si ha $7^{18} = 4$.

MCD e algoritmi di Euclide ed Euclide esteso

Definizione

Il **massimo comun divisore** di due numeri $a, b \in \mathbb{N}$, non entrambi nulli, è il loro *più grande divisore comune* ed è notato $\text{MCD}(a, b)$.

Il massimo comun divisore possiede le seguenti proprietà:

- 1 $\text{MCD}(a, a) = a$ (se $a > 0$);
- 2 $\text{MCD}(a, 0) = a$;
- 3 $\text{MCD}(a, b) = \text{MCD}(a - b, b)$, se $a > b$.
- 4 $\text{MCD}(a, b) = \text{MCD}(b, r)$, se $a > b$ ed r è il resto della divisione euclidea $a = bq + r$.

Definizione

Due numeri naturali a, b sono detti **coprimi** (o anche relativamente primi) se $\text{MCD}(a, b) = 1$.

MCD e algoritmi di Euclide ed Euclide esteso

Un algoritmo, noto come **algoritmo di Euclide**, permette di calcolare rapidamente l'MCD tra due numeri (grandi) sulla base della proprietà $\text{MCD}(a, b) = \text{MCD}(b, r)$.

Algoritmo di Euclide

- START $k = 1$:
 - porre $a_1 = a$ e $b_1 = b$ con $a > b$;
 - calcolare r_1 tale che $a_1 = b_1q_1 + r_1$;
- PASSO $k = 2$:
 - porre $a_2 = b_1$ e $b_2 = r_1$;
 - calcolare r_2 tale che $a_2 = b_2q_2 + r_2$;
- PASSO k generale:
 - porre $a_k = b_{k-1}$ e $b_k = r_{k-1}$;
 - calcolare r_k tale che $a_k = b_kq_k + r_k$;
- STOP : quando $r_n = 0$ allora $b_n = r_{n-1} = \text{MCD}(a, b)$.

MCD e algoritmi di Euclide ed Euclide esteso

Ecco un esempio: $a = 123$, $b = 14$.

k	a_k	b_k	r_k	
1	123	14	11	$123 = 14 \cdot 8 + 11$
2	14	11	3	$14 = 1 \cdot 11 + 3$
3	11	3	2	$11 = 3 \cdot 3 + 2$
4	3	2	1	$3 = 1 \cdot 2 + 1$
5	2	1	0	$2 = 1 \cdot 2 + 0$

Quindi $\text{MCD}(123, 14) = 1$ e i numeri 123 e 14 sono *coprimi*.

Osserviamo che $123 = 3 \cdot 41$, $14 = 7 \cdot 2^2$ e 11 è primo, e si ha

$$\begin{aligned} \text{MCD}(123, 14) &= \text{MCD}(14, 11) = \text{MCD}(11, 3) = \text{MCD}(3, 2) \\ &= \text{MCD}(2, 1) = \text{MCD}(1, 0) = 1. \end{aligned}$$

MCD e algoritmi di Euclide ed Euclide esteso

Il seguente teorema permette di scrivere l'MCD in modo utile al calcolo dell'inverso in \mathbb{Z}_n .

Teorema

Siano $a, b \in \mathbb{Z}$, allora esistono $s, t \in \mathbb{Z}$ tale che

$$as + bt = \text{MCD}(a, b) .$$

- Se a e b sono coprimi si ha l'**identità di Bézout**

$$as + bt = 1 .$$

- Un algoritmo noto come **algoritmo di Euclide esteso** permette di calcolare s, t e $\text{MCD}(a, b)$.
- I numeri $s, t \in \mathbb{Z}$ non sono unici. Ad esempio

$$3 = \text{MCD}(9, 6) = 1 \cdot 9 + (-1) \cdot 6 = 9 \cdot 3 + (-4) \cdot 6 .$$

MCD e algoritmi di Euclide ed Euclide esteso

Algoritmo di Euclide esteso

- START $k = 1$: porre (con $a > b$)

$$r_0 = a, s_0 = 1, t_0 = 0 \text{ e } r_1 = b, s_1 = 0, t_1 = 1;$$

- PASSO $k = 2$:

- calcolare q_1 e r_2 tale che $r_0 = r_1q_1 + r_2$;

- calcolare $\begin{cases} s_2 = s_0 - s_1q_1 \\ t_2 = t_0 - t_1q_1 \end{cases}$

- PASSO k generale:

- calcolare q_{k-1} e r_k tale che $r_{k-2} = r_{k-1}q_{k-1} + r_k$;

- calcolare $\begin{cases} s_k = s_{k-2} - s_{k-1}q_{k-1} \\ t_k = t_{k-2} - t_{k-1}q_{k-1} \end{cases}$

- STOP : quando $r_{n+1} = 0$, allora $r_n = as_n + bt_n$ e

$$r_n = \text{MCD}(a, b) \quad s_n = s \quad t_n = t .$$

MCD e algoritmi di Euclide ed Euclide esteso

Ecco un esempio: $a = 123$, $b = 14$.

k	r_k	s_k	t_k	q_k	
0	123	1	0	—	
1	14	0	1	8	$123 = 8 \cdot 14 + 11$
2	11	1	-8	1	$14 = 1 \cdot 11 + 3$
3	3	-1	9	3	$11 = 3 \cdot 3 + 2$
4	2	4	-35	1	$3 = 1 \cdot 2 + 1$
5	1	-5	44	2	$2 = 1 \cdot 2 + 0$

Quindi $\text{MCD}(123, 14) = 1$, $s = -5$ e $t = 44$, da cui

$$1 = 123 \cdot (-5) + 14 \cdot 44 .$$

L'inverso in \mathbb{Z}_n

Definizione

Sia $a \in \mathbb{Z}$ e $n \in \mathbb{N}$, $n > 1$, tali che $\text{MCD}(a, n) = 1$. Un elemento $\tilde{a} \in \mathbb{Z}_n$ tale che $a\tilde{a} \equiv 1 \pmod{n}$ è detto **inverso di a modulo n** .
Ossia $a\tilde{a} = 1$ in \mathbb{Z}_n .

- La congruenza equivale a

$$a\tilde{a} + n(-k) = 1 = \text{MCD}(a, n)$$

è quindi possibile calcolare \tilde{a} (e $(-k)$) con l'algoritmo di Euclide esteso (se il il valore di $\tilde{a} \notin \mathbb{Z}_n$ si torva il suo elemento congruente in \mathbb{Z}_n).

- In generale **non** tutti gli elementi (non nulli) di \mathbb{Z}_n hanno un inverso.

L'inverso in \mathbb{Z}_n

Ecco ad esempio le tavole di moltiplicazione di \mathbb{Z}_7 e \mathbb{Z}_9 .

\cdot	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

\cdot	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8
2	0	2	4	6	8	1	3	5	7
3	0	3	6	0	3	6	0	3	6
4	0	4	8	3	7	2	6	1	5
5	0	5	1	6	2	7	3	8	4
6	0	6	3	0	6	3	0	6	3
7	0	7	5	3	1	8	6	4	2
8	0	8	7	6	5	4	3	2	1

Esempio: l'inverso di 14 in \mathbb{Z}_{123} .

L'*algoritmo di Euclide esteso* fornisce

$$1 = 123 \cdot (-5) + 14 \cdot 44 \iff 14 \cdot 44 - 1 = 5 \cdot 123 \iff 14 \cdot 44 \equiv 1 \pmod{123}$$

Quindi $14 \cdot 44 = 1$ in \mathbb{Z}_{123} e l'inverso cercato è 44.

La funzione $\varphi(n)$ di Euler

Definizione

La **funzione indicatrice di Euler** è l'applicazione φ che ad ogni numero naturale $n > 1$ associa il numero di numeri naturali inferiori a n e coprimi a n :

$$\varphi(n) = \#\{a \in \mathbb{N} : a \leq n \text{ e } \text{MCD}(a, n) = 1\} .$$

$\varphi(n)$ indica quindi il numero di elementi invertibili in \mathbb{Z}_n .

La funzione indicatrice di Euler possiede le seguenti proprietà:

- ❶ $\varphi(p) = p - 1$, se p è un numero primo;
- ❷ $\varphi(p^r) = p^{r-1}(p - 1)$, se p è un numero primo e $r \in \mathbb{N}^*$;
- ❸ $\varphi(nm) = \varphi(n)\varphi(m)$, se $\text{MCD}(n, m) = 1$ e $n, m \in \mathbb{N}^*$.

Ad esempio: $\varphi(7) = 7 - 1 = 6$ e $\varphi(9) = 3^1(3 - 1) = 6$.

Il piccolo teorema di Fermat e il teorema di Euler

Piccolo teorema di Fermat

- Sia $a \in \mathbb{N}$ e p un numero primo. Allora $a^p \equiv a \pmod{p}$.
- Sia $a \in \mathbb{N}$ e p un numero primo tali che $\text{MCD}(a, p) = 1$. Allora $a^{p-1} \equiv 1 \pmod{p}$.

Il piccolo teorema di Fermat fornisce velocemente l'inverso di a in \mathbb{Z}_p nel caso in cui p è primo. Infatti

$$aa^{p-2} \equiv 1 \pmod{p}$$

ossia a^{p-2} è (congruente al)l'inverso cercato.

Ad esempio abbiamo

$$4^{7-1} \equiv 1 \pmod{7} \implies 4^{7-1} = 1 \text{ in } \mathbb{Z}_7$$

e $4^{7-2} \equiv 2 \pmod{7}$ cioè 2 è l'inverso di 4 in \mathbb{Z}_7 .

Il piccolo teorema di Fermat e il teorema di Euler

Teorema di Euler

Siano $a, n \in \mathbb{N}$, $n > 1$, tali che $\text{MCD}(a, n) = 1$. Allora
 $a^{\varphi(n)} \equiv 1 \pmod{n}$.

Grazie al teorema di Euler si può osservare che (se esiste) l'inverso di a in \mathbb{Z}_n è dato da $a^{\varphi(n)-1}$ (o il suo congruente in \mathbb{Z}_n), infatti si ha

$$aa^{\varphi(n)-1} \equiv 1 \pmod{n}.$$

Ad esempio abbiamo

$$2^{\varphi(9)} \equiv 1 \pmod{9} \implies 2^{\varphi(9)} = 1 \text{ in } \mathbb{Z}_9$$

e $2^{\varphi(9)-1} \equiv 5 \pmod{9}$ cioè 5 è l'inverso di 2 in \mathbb{Z}_9 .

Un complemento sulle strutture algebriche

Definizione

Sia (E, \star, \diamond) un anello commutativo. Un elemento $a \in E$, $a \neq 0$, è detto **divisore di zero** se esiste un elemento $b \in E$, $b \neq 0$, tale che

$$a \diamond b = 0$$

In questo caso anche b è un divisore di zero.

Ad esempio nell'anello $(\mathbb{Z}_9, +, \cdot)$ abbiamo

$$3 \cdot 6 = 0$$

3 e 6 sono quindi divisori di zero in \mathbb{Z}_9 .

Teorema

Se p è un numero primo allora \mathbb{Z}_p non ammette divisori di zero.

Un complemento sulle strutture algebriche

Definizione

Un anello commutativo (E, \star, \diamond) privo di divisori di zero è detto **anello integro**.

- In un anello integro tutti gli elementi diverso da 0 sono semplificabili rispetto a \diamond , ossia

$$a \diamond b = a \diamond c \implies b = c$$

$$b \diamond a = c \diamond a \implies b = c$$

- Questo non è vero se l'anello ammette dei divisori di zero.

Ad esempio in $(\mathbb{Z}_9, +, \cdot)$

$$3x = 3y \not\Rightarrow x = y$$

cosa che invece è vera in $(\mathbb{Z}_7, +, \cdot)$

$$3x = 3y \implies x = y .$$

Un complemento sulle strutture algebriche

Definizione

Siano (E, \star, \diamond) un anello commutativo. Se per ogni $a \in E$, $a \neq 0$, esiste l'elemento inverso, allora (E, \star, \diamond) è detto un **campo**.

Teorema

Se p è un numero primo allora \mathbb{Z}_p è un campo.

La notazione posizionale

Nel sistema *decimale*, il numero 4634 si può scrivere come

$$4634 = 4 \cdot 10^3 + 6 \cdot 10^2 + 3 \cdot 10^1 + 4 \cdot 10^0$$

- Le posizioni delle cifre, da destra a sinistra, corrispondono alle potenze di 10. La prima posizione è quella a potenza zero.
- Il valore numerico è la somma delle moltiplicazioni della cifra nella posizione i -esima per 10^i .

Questo sistema di scrivere i valori numerici è detto **notazione posizionale**: i simboli (cifre) usati per scrivere i numeri assumono valori diversi a seconda della *posizione* che occupano nella notazione.

La notazione posizionale

Ecco una definizione di sistema di numerazione posizionale:

- si sceglie un numero naturale $b \geq 2$, che chiameremo **base**;
- si scelgono n numeri $(\beta_{n-1}\beta_{n-2}\dots\beta_1\beta_0)_b$, che chiameremo **cifre** dove $\beta_i \in \mathbb{N}$ e $0 \leq \beta_i \leq b - 1$;
- si compongono i numeri tenendo presente che il valore di ogni cifra va moltiplicato per:
 - b^0 cioè 1 (unità) se è l'ultima cifra alla destra del numero che stiamo considerando;
 - b^1 cioè b se è la seconda cifra da destra;
 - $b^{(i-1)}$ se è la i -esima cifra da destra;
- la somma tutti i valori così ottenuti è il *numero decimale* che stiamo considerando:

$$M \equiv (M)_{10} = \sum_{i=0}^{n-1} \beta_i b^i .$$

Cambiamento di base: dalla base b alla base 10

Dato un numero in base b , ossia la sequenza $(\beta_{n-1}\beta_{n-2}\dots\beta_1\beta_0)_b$ delle sue cifre, si ottiene il numero in base 10 scrivendo in *forma polinomiale*

$$(M)_{10} = \sum_{i=0}^{n-1} \beta_i b^i .$$

Ad esempio le cifre $(1100)_2$ in base 2 danno il numero seguente in base 10

$$1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = (12)_{10}$$

Quindi

$$(1100)_{\text{base } 2} = (12)_{\text{base } 10}$$

Cambiamento di base: dalla base 10 alla base b

Dato un numero in base 10 $x \equiv (x)_{10}$, ossia la sequenza $(x_{n-1}x_{n-2} \dots x_1x_0)_{10}$ delle sue cifre, si ottiene il numero in base b con le seguenti divisioni euclidee:

- $x = q_0b + r_0$;
- $q_0 = q_1b + r_1$;
- $q_1 = q_2b + r_2$;
- ...
- $q_{n-2} = q_{n-1}b + r_{n-1}$ dove $q_{n-1} = 0$.

I resti ottenuti, *letti dall'ultimo al primo*, sono le cifre del numero $(x)_{10}$ in base b : $(r_{n-1}r_{n-2} \dots r_1r_0)_b$, ossia

$$(x)_{10} = (x_{n-1}x_{n-2} \dots x_1x_0)_{10} = (r_{n-1}r_{n-2} \dots r_1r_0)_b .$$

Cambiamento di base: dalla base 10 alla base b

Esempio $(4634)_{10}$ in base 10

- $4634 = 463 \cdot 10 + 4;$
- $463 = 46 \cdot 10 + 3;$
- $46 = 4 \cdot 10 + 6;$
- $4 = 0 \cdot 10 + 4;$

e si ottengono le cifre attese.

Esempio $(12)_{10}$ in base 2

- $12 = 6 \cdot 2 + 0;$
- $6 = 3 \cdot 2 + 0;$
- $3 = 1 \cdot 2 + 1;$
- $1 = 0 \cdot 2b + 1.$

Quindi $(12)_{10} = (1100)_2.$

La scomposizione in base 2 dell'esponente di una potenza in \mathbb{Z}_n è utile per l'algoritmo di Legendre.

Attenzione: Nelle applicazioni un numero decimale sarà scritto senza precisare la base 10 $(\cdot)_{10}$.

Il logaritmo e la lunghezza di un numero

Definizione

Siano $a, b \in \mathbb{R}_+$, $a \neq 1$. Allora $x \in \mathbb{R}$ è il **logaritmo di base a** di b se

$$a^x = b \iff x = \log_a(b).$$

Le proprietà del logaritmo utili per i nostri scopi sono:

- ① $\log_a(a) = 1$;
- ② $\log_a(xy) = \log_a(x) + \log_a(y)$;
- ③ $\log_a(x^r) = r \log_a(x)$.

Ad esempio

$$\begin{aligned} 4807 = 4.807 \cdot 10^3 &\implies \log_{10}(4807) = \log_{10}(4.807) + \log_{10}(10^3) \\ &= \log_{10}(4.807) + 3 \log_{10}(10) \\ &= 0.68 + 3 = 3.68 \end{aligned}$$

Il logaritmo e la lunghezza di un numero

- Per “misurare” la lunghezza di un numero decimale M “contiamo” le sue cifre.
- Il logaritmo di base 10 permette di ottenere questo dato rapidamente

$$\# \text{ cifre } M = \lfloor \log_{10}(M) + 1 \rfloor$$

dove $\lfloor x \rfloor$ è la parte intera di x .

Ad esempio

$$\# \text{ cifre } 4807 = \lfloor \log_{10}(4807) + 1 \rfloor = \lfloor 3.68 + 1 \rfloor = \lfloor 4.68 \rfloor = 4$$

I numeri primi

Definizione

Un numero $p \in \mathbb{N}$, $p > 1$, è detto **numero primo** se $a|p$, con $a \in \mathbb{N}$, implica $a = p$ o $a = 1$, ossia se i soli divisori di p in \mathbb{N} sono 1 e se stesso. I numeri in \mathbb{N} maggiori o uguali a 2 che non sono primi sono detti numeri composti.

- I numeri primi sono i “mattoni” di base dell'aritmetica.
- Il numero primo più grande oggi conosciuto è detto il **numero di Mersenne**

$$M_{43112609} = 2^{43112609} - 1$$

che è della forma $M_p = 2^p - 1$ con p a sua volta primo.

Teoremi sui numeri primi

Teorema fondamentale dell'aritmetica

Ogni numero naturale $n > 1$ può essere scritto come il prodotto finito di numeri primi (chiamati fattori primi), e questa espressione è unica (salvo l'ordine), ossia

$$n = p_1^{m_1} p_2^{m_2} \cdots p_r^{m_r}$$

dove p_1, \dots, p_r sono numeri primi e $m_1, \dots, m_r \in \mathbb{N}^*$.

Teorema di Euclide

Esistono infiniti numeri primi.

Teorema

Se $n > 1$ è un numero naturale composto, allora possiede un fattore primo inferiore o uguale a \sqrt{n} .

Alcuni criteri di primalità

Per testare se un numero $n \in \mathbb{N}$ *non* è primo è possibile utilizzare per esempio il *piccolo teorema di Fermat*, infatti

$$(\forall a \in \mathbb{N} \text{ e } p \text{ **primo**} \implies a^p \equiv a \pmod{p})$$



$$(\exists a \in \mathbb{N} \text{ e } a^p \not\equiv a \pmod{p} \implies p \text{ **non primo**})$$

È importante osservare che $(\forall a \in \mathbb{N} \text{ e } a^p \equiv a \pmod{p} \not\implies p \text{ **primo**})$.

Ad esempio

- $3^{27} \equiv 0 \pmod{27}$ quindi 27 *non* è primo;
- $5^{11} \equiv 5 \pmod{11} \rightarrow$ non si può concludere;
- $5^{15} \equiv 5 \pmod{15} \rightarrow$ non si può concludere;
- $7^{15} \equiv 13 \pmod{15} \rightarrow$ quindi 15 *non* è primo.

Esistono numeri, detti **numeri di Carmichael**, per i quali, per ogni a , $a^n \equiv a \pmod{n}$ **ma** n è composto (ad esempio: $3 \cdot 11 \cdot 17 = 561$).

Alcuni criteri di primalità

Il piccolo teorema di Fermat permette di ottenere un semplice esempio di **test di primalità probabilistico**. Si procede così:

- si sceglie un numero a aleatoriamente;
- si testa se $a^n \equiv a \pmod{n}$, se la congruenza non è verificata *stop* e n è composto (a è detto *testimone* contro la primalità);
- si ripete l'operazione fino ad avere la certezza richiesta, in tal caso se n non è stato riconosciuto come composto è detto *probabilmente primo*.

Chiaramente i numeri di Carmichael sfuggono al test per ogni a .

Attenzione: sapere se n è composto non implica conoscere la sua fattorizzazione.

Alcuni criteri di primalità

Un altro criterio di primalità, più di interesse teorico che altro, è il teorema seguente

Teorema (di Wilson)

Un numero naturale $n > 1$ è primo se e solo se n divide $(n - 1)! + 1$.

dove $n! = n \cdot (n - 1) \cdot (n - 2) \cdot 2 \cdot 1$.

Ad esempio

- $n = 11$ si ha $(11 - 1)! + 1 = 3628801$ e $11 \mid 3628801$, 11 è primo;
- $n = 15$ si ha $(15 - 1)! + 1 = 87178291201$ e $15 \nmid 87178291201$, 15 non è primo.

Rarefazione dei numeri primi

Teorema di rarefazione dei numeri primi

Se n è un numero naturale, il numero di numeri primi tra 1 e n è approssimativamente uguale a

$$\#\{m \in \mathbb{N} : 1 < m \leq n, m \text{ primo}\} \approx \frac{n}{\ln n}.$$

Dove $\ln(x) = \log_e(x)$ con $e = 2.718281828\dots \in \mathbb{R} \setminus \mathbb{Q}$ chiamato *numero di Euler*.

La fattorizzazione dei numeri primi

Sia $n = ab$, stimiamo il numero di tentativi per scoprire a e b usando il teorema della radice quadrata.

- Sappiamo che un fattore (ad esempio a) soddisfa $a \leq \sqrt{n}$ allora vi sono approssimativamente

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

numeri primi da testare come divisori di n .

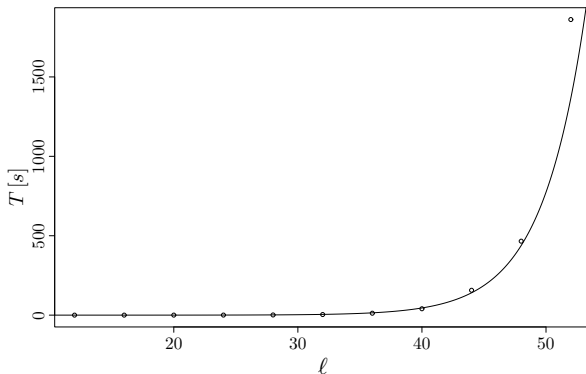
- Se $n \approx 10^{50}$ (quindi un numero di 50 cifre) allora $\sqrt{n} \approx 10^{25}$ e i tentativi sono

$$\frac{\sqrt{n}}{\ln \sqrt{n}} \approx 1.74 \cdot 10^{23} .$$

- Per un computer capace di effettuare 1000 miliardi di test al secondo sono necessari più di $1.7 \cdot 10^{11}$ secondi, ossia circa 5600 anni!

La fattorizzazione dei numeri primi

Prova di fattorizzazione con MAPLE 9 su PENTIUM 4, 2.60 GHz.



Tempo di fattorizzazione T in funzione della lunghezza l del numero

$$T = 0.0005e^{0.285l} .$$

La fattorizzazione dei numeri primi

- Dal risultato precedente per $\ell = 300$ (RSA) si ottiene

$$T \approx 10^{26} \text{ y} > \text{età dell'Universo} .$$

- Anche per gli algoritmi e i computer più veloci il problema di fattorizzare in numeri primi un numero con molte cifre è praticamente impossibile da risolvere.

Il problema della fattorizzazione:

- *non* è (per ora) *risolvibile* in un *tempo polinomiale* in ℓ , bensì in un *tempo esponenziale*;
- data la soluzione la *verifica* si ottiene in un *tempo polinomiale*.

La fattorizzazione dei numeri primi

Secondo la teoria della **complessità computazionale**, la teoria che studia le risorse di calcolo richieste per risolvere un dato problema, vi sono:

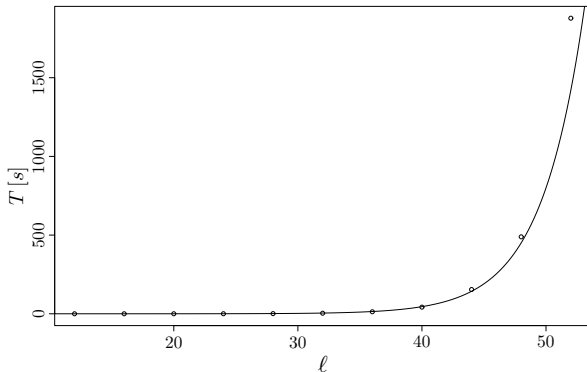
- problemi **semplici** (detti della classe **P**): esiste un algoritmo che richiede un **tempo polinomiale** rispetto alla dimensione dei dati del problema (il numero di cifre nella fattorizzazione);
- problemi **difficili** (detti della classe **NP**), ossia problemi per i quali data una possibile soluzione la sua **verifica** avviene con un algoritmo **in tempo polinomiale**.

Per la classe **NP** esistono algoritmi **esponenziali** . . . ma non si sa se esistono algoritmi polinomiali.

La fattorizzazione è un problema **NP** ed è questa la “garanzia” della sicurezza della crittografia RSA.

La fattorizzazione dei numeri primi

Un altro problema della classe **NP** è il calcolo della *funzione* $\varphi(n)$ di Euler senza conoscere la fattorizzazione $n = pq$.



Tempo di calcolo T in funzione della lunghezza l del numero

$$T = 0.0005e^{0.287l} .$$

Crittografia a chiave pubblica

- Il destinatario \mathbb{D} pubblica la propria **chiave pubblica** e ogni mittente \mathbb{M} che desidera inviargli un messaggio dovrà usarla per cifrare il proprio messaggio.
- La chiave pubblica di \mathbb{D} è nota a tutti, come pure il messaggio cifrato inviato da \mathbb{M} .
- Solo \mathbb{D} sa come decifrarlo grazie alla sua **chiave privata**, tale chiave è ignota anche a \mathbb{M} .

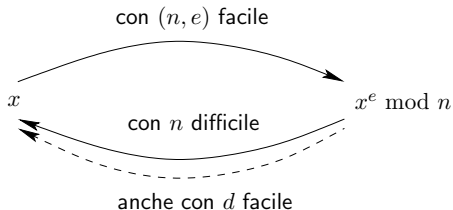
Questo tipo di crittografia è noto come **crittografia a chiave asimmetria** o **crittografia a chiave pubblica**, poiché \mathbb{M} e \mathbb{D} non condividono una chiave privata segreta e comune.

Crittografia a chiave pubblica

Il principio della crittografia a chiave pubblica si basa sul fatto che la conoscenza della chiave pubblica e dell'algoritmo stesso non siano sufficienti per risalire alla chiave privata. Tale meccanismo è reso possibile grazie all'uso di **funzioni unidirezionali (a botola)**:

- facili da calcolare;
- difficili da invertire;
- data un'informazione supplementare (la botola) la funzione è facile da invertire.

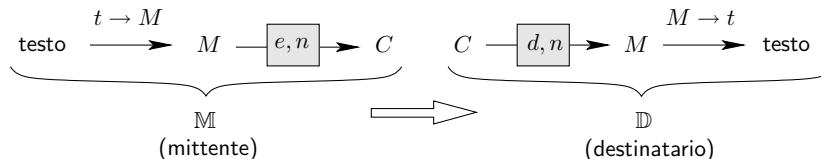
Ad esempio:



Il protocollo RSA

Il protocollo di crittografia RSA è diviso nelle seguenti parti:

- la generazione delle chiavi pubblica (e, n) e privata (d, n) ;
- la conversione del testo in un numero M e viceversa;
- la cifratura del messaggio M e la sua decifratura.



La generazione delle chiavi

- Si scelgono aleatoriamente due **numeri primi** p e q e si calcola $n = pq$
(in pratica n ha circa *300 cifre decimali* ($n \approx 10^{300}$), p e q non sono troppo vicini, $(p - 1)$ e $(q - 1)$ devono avere fattori primi grandi);
- si calcola la funzione di Euler

$$\varphi(n) = \varphi(pq) = \varphi(p)\varphi(q) = (p - 1)(q - 1) ;$$

- si sceglie $e < \varphi(n)$ coprimo a $\varphi(n)$, ossia $\text{MCD}(e, \varphi(n)) = 1$;
- si calcola d tale che $ed \equiv 1 \pmod{\varphi(n)}$, ossia d è l'inverso di e in $\mathbb{Z}_{\varphi(n)}$.

La condizione $\text{MCD}(e, \varphi(n)) = 1$ garantisce l'esistenza dell'inverso di e in $\mathbb{Z}_{\varphi(n)}$.

- La coppia (e, n) costituisce la **chiave pubblica**;
- la coppia (d, n) costituisce la **chiave privata**.

La generazione delle chiavi

Ecco un esempio:

- scegliamo $p = 11$, $q = 5$, quindi $n = 55$;
- si ha $\varphi(55) = \varphi(11)\varphi(5) = 40$;
- scegliamo $e = 7$, poiché si ha $\text{MCD}(7, 40) = 1$;
- si ottiene $d = 23$, infatti $7 \cdot 23 (= 161) \equiv 1 \pmod{40}$.

- **Chiave pubblica:** $(e = 7, n = 55)$.
- **Chiave privata:** $(d = 23, n = 55)$.

Dal testo al numero e viceversa

È necessario definire un alfabeto in cui ogni lettera è messa in corrispondenza con un numero.

- Consideriamo l'**alfabeto** seguente (spazio, lettere minuscole) composto da 27 elementi

$\{ \text{ , } a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z \}$

messi in corrispondenza con i 27 numeri

$\{0, 1, 2, 3, \dots, 24, 26\}$.

- Il messaggio di k lettere $\alpha_{k-1} \dots \alpha_0$ è in corrispondenza con la sequenza di numeri $\ell_{k-1} \dots \ell_0$

$\alpha_{k-1} \dots \alpha_0 \longleftrightarrow \ell_{k-1} \dots \ell_0$.

Dal testo al numero

- Si stabilisce la corrispondenza $\alpha_{k-1} \dots \alpha_0 \leftrightarrow \ell_{k-1} \dots \ell_0$;
- si considera la sequenza $\ell_{k-1} \dots \ell_0$ come le cifre di un numero M in **base 27** ossia $M = (\ell_{k-1} \dots \ell_0)_{27}$;
- si trasforma $(\ell_{k-1} \dots \ell_0)_{27}$ in **base 10**

$$M = \sum_{i=0}^{k-1} \ell_i 27^i .$$

Ad esempio: ciao

$$\alpha_3 = \mathbf{c}, \alpha_2 = \mathbf{i}, \alpha_1 = \mathbf{a}, \alpha_0 = \mathbf{o} \leftrightarrow \ell_3 = 3, \ell_2 = 9, \ell_1 = 1, \ell_0 = 15$$

quindi

$$M = (3\ 9\ 1\ 15)_{27} \implies M = 15 \cdot 27^0 + 1 \cdot 27^1 + 9 \cdot 27^2 + 3 \cdot 27^3 = 65652$$

Dal numero al testo

- Si effettuano le divisioni euclidee successive per ottenere le cifre in **base 27** del numero M in **base 10**, ossia $M = (\ell_{k-1} \dots \ell_0)_{27}$;
- si ristabilisce la corrispondenza $\ell_{k-1} \dots \ell_0 \leftrightarrow \alpha_{k-1} \dots \alpha_0$.

Ad esempio: $M = 65652$

$$65652 = 2431 \cdot 27 + 15$$

$$2431 = 90 \cdot 27 + 1$$

$$90 = 3 \cdot 27 + 9$$

$$3 = 0 \cdot 27 + 3$$

da cui $M = (3\ 9\ 1\ 15)_{27}$ e si ritrova il messaggio ciao.

Cifrare e decifrare

Dato un messaggio

$$1 < M < n \quad (\text{i calcoli sono in } \mathbb{Z}_n)$$

(se $M \geq n$ è necessario suddividere il messaggio in più messaggi di lunghezza adeguata).

- Il **mittente** \mathbb{M} cifra M grazie a $\mathcal{P} = (e, n)$ come segue

$$E_{\mathcal{P}}(M) \rightarrow C \equiv M^e \pmod{n},$$

C costituisce il messaggio cifrato di M che viene inviato al destinatario \mathbb{D} ;

- il **destinatario** \mathbb{D} decifra C grazie a $\mathcal{Q} = (d, n)$

$$D_{\mathcal{Q}}(C) \rightarrow M \equiv C^d \pmod{n}.$$

Cifrare e decifrare

Ecco un esempio.

Sia il messaggio $M = 13$ e la chiave dell'esempio precedente:

$$\mathcal{P} = (e = 7, n = 55) \quad \text{e} \quad \mathcal{Q} = (d = 23, n = 55)$$

- Il **mittente** \mathbb{M} cifra M

$$E_{\mathcal{P}}(13) \rightarrow 7 \equiv 13^7 \pmod{55},$$

$C = 7$ costituisce il messaggio cifrato di $M = 13$ che viene inviato al destinatario \mathbb{D} ;

- il **destinatario** \mathbb{D} decifra C

$$D_{\mathcal{Q}}(7) \rightarrow 13 \equiv 7^{23} \pmod{55}.$$

Il teorema RSA

Alla base del sistema di crittografia a chiave pubblica RSA (dal nome dei tre matematici Rivest, Shamir e Adleman che la inventarono nel 1977) vi è il seguente teorema che riportiamo per completezza

Teorema RSA

Siano p e q sono due numeri primi distinti e $n = pq$, se e è coprimo a $\varphi(n)$ e d è tale che $ed \equiv 1 \pmod{\varphi(n)}$, allora per ogni $a \in \mathbb{N}$

$$a^{ed} \equiv a \pmod{n} .$$

La sicurezza di RSA

- Per **rompere il codice RSA** è necessario ottenere d della chiave privata;
- è quindi necessario conoscere $\varphi(n)$.

Osserviamo che da $\varphi(n)$ e n si ricavano *facilmente* p e q :

$$\varphi(n) = (p-1)(q-1) = n - (p+q) + 1 \implies \begin{cases} p + q &= n - \varphi(n) + 1 \\ pq &= n \end{cases}$$

quindi p e q sono le soluzioni dell'equazione di secondo grado (relazioni di Viète)

$$x^2 - (p + q)x + pq = 0 .$$

- Dal punto di vista della **complessità computazionale** trovare $\varphi(n)$ *senza* la fattorizzazione $n = pq$ è equivalente alla fattorizzazione (entrambi hanno un algoritmo esponenziale);
- ma la fattorizzazione è un problema difficile (tipo **NP**) per i numeri grandi e questa è la **sicurezza di RSA**.

La sicurezza di RSA

La *RSA Security* pubblica sfide per rompere il codice RSA, dette *RSA- X* , dove X è il numero di cifre decimali di n o di bit (cifre in base 2) del numero binario associato al numero decimale n .

- **RSA-129** – 129 cifre decimali (prima sfida, 1977):
fattorizzato nel 1994 in 8 mesi con 600 computer in rete;
- **RSA-640** – 193 cifre decimali: fattorizzato nel 2005;
- **RSA-200** – 200 cifre decimali: fattorizzato nel 2005 in 3 mesi su un cluster di 80 CPU di 2.2 GHz, ecco i numeri

$$\begin{aligned}
 n &= 2799783391122132787082946763872260162107044678695542853756000992932 \\
 &6128400107609345671052955360856061822351910951365788637105954482006 \\
 &576775098580557613579098734950144178863178946295187237869221823983 \\
 p &= 35324619344027701212726049781984643686711974001976 \\
 &25023649303468776121253679423200058547956528088349 \\
 q &= 79258699544783330333470858414800596877379758573642 \\
 &19960734330341455767872818152135381409304740185467
 \end{aligned}$$

Alcune sfide aperte: **RSA-704** – 212 cifre decimali (30'000 \$),
RSA-1024 – 309 cifre decimali (100'000 \$).

La firma digitale

La crittografia RSA può essere utilizzata come **firma digitale**.

Sia M il messaggio (tutti possono vederlo ma **il destinatario deve avere la certezza che il mittente sia quello che si è firmato**).

- Il **mittente** \mathbb{M} firma M grazie alla sua chiave privata $\mathcal{Q} = (d, n)$ come segue

$$F_{\mathcal{Q}}(M) \rightarrow C \equiv M^d \pmod{n},$$

- il **destinatario** \mathbb{D} decifra C grazie alla chiave pubblica $\mathcal{P} = (e, n)$ del mittente \mathbb{M}

$$D_{\mathcal{P}}(C) \rightarrow M \equiv C^e \pmod{n}.$$

Tutti possono decifrare C ma vi è la garanzia che il mittente è uno solo, infatti solo il mittente autentico possiede la chiave privata e associata a quella pubblica d utilizzata per decifrare.

La firma digitale e il messaggio segreto

È possibile avere la segretezza e la firma digitale, ossia:

- solo il destinatario \mathbb{D} può decodificare il messaggio (**segreto**);
- solo il mittente \mathbb{M} può averlo inviato (**firma digitale**).

Ecco come si procede.

- il mittente firma il messaggio con la propria chiave privata:

$$F \equiv M^{d_{\mathbb{M}}} \pmod{n_{\mathbb{M}}};$$
- il mittente codifica F con la chiave pubblica di \mathbb{D} :

$$C \equiv F^{e_{\mathbb{D}}} \pmod{n_{\mathbb{D}}};$$
- il destinatario decodifica C con la propria chiave privata:

$$F \equiv C^{d_{\mathbb{D}}} \pmod{n_{\mathbb{D}}};$$
- il destinatario decodifica F con la chiave pubblica di \mathbb{M} :

$$M \equiv F^{e_{\mathbb{M}}} \pmod{n_{\mathbb{M}}}.$$