NUMERI PRIMI E CRITTOGRAFIA

A cura di

Marco Evangelista Valentina Testa Maura Tuzzolo

sotto la guida della Prof.ssa Francesca Tovena

NUMERI PRIMI E CRITTOGRAFIA

Obiettivi del laboratorio

Lo studio della matematica rappresenta uno scoglio per la gioventù contemporanea. Tale disciplina viene concepita come un qualcosa di astratto e inutile in cui solo pochi possono riuscire. Per superare questa impostazione abbiamo deciso di analizzare gli aspetti matematici della crittografia: la necessità di comunicare in segretezza è un bisogno reale e ognuno di noi, nel suo piccolo, ne ha fatto esperienza. Questo è il punto di partenza del nostro lavoro: sviluppare teorie matematiche non è un vezzo fine a se stesso ma trova applicazioni concrete.
L'analisi dei vari sistemi crittografici offre lo spunto per introdurre teorie matematiche che gli studenti non hanno mai analizzato: l'algebra modulare e la teoria dei numeri primi.
Questo è il nodo fondamentale del laboratorio: creare "nuova" matematica per migliorare e ampliare le conoscenze degli studenti.
Nell'ambito dell'algebra modulare trattiamo l'Algoritmo Euclideo, l'Identità di Bezout, le congruenze, le caratteristiche dell'anello Z_n (definizione delle operazioni, degli elementi neutri, ecc.), la risoluzione delle equazioni alle congruenze.
La teoria dei numeri primi, invece, viene affrontata con meno rigore (anche perché è ancora terreno fertile di studi) cercando di fornire ai ragazzi un'idea intuitiva della loro importanza, di come riconoscerli e di come costruirli.
Infine accenniamo al problema della scomposizione di un numero in primi specificando che, in linea teorica, essa è sempre possibile ma che, in pratica, è difficile da realizzare a causa dei tempi elevati che richiede.

Aspetti significativi del programma

L'introduzione di nuovi impianti teorici può rappresentare un problema se non si capisce profondamente la loro utilità. Ogni nuova struttura matematica che introdurremo, quindi, è presentata in relazione al suo utilizzo e alla sua efficacia.
Escluso il concetto di congruenza che è presentato come modellatura matematica di quanto si mette in atto con la cifratura per traslazione, ogni altra nuova nozione è presentata o come soluzione di un problema sorto nell' ambito che stiamo studiando o come un miglioramento pratico della tecniche che già possediamo.
Infatti, una volta affrontato il tema delle congruenze (lez. 1), si porrà il problema di trovare un metodo di cifratura più sofisticato che non solo trasli le lettere ma che le permuti. Per realizzare ciò sarà necessario introdurre la struttura di Z_n e le equazioni alla congruenza, risolubili grazie all'Algoritmo di Euclide e all'Identità di Bezout (lez. 2-3-4).
Notando che anche questo sistema di cifratura è violabile si introdurrà il concetto di funzione unilaterale e di cifratura a chiave pubblica. A questo punto si enuncerà il sistema RSA dando ampio spazio all'assetto matematico che lo sostiene spiegando approfonditamente la funzione di Eulero, e i teoremi ad essa collegati, e sottolineando l'importanza del sistema di numerazione binaria che consente di agevolare i conti (lez. 5-6).
Si tratterà, infine, la teoria dei numeri primi e della fattorizzazione, in quanto sugli aspetti costitutivi di queste si fonda la validità teorica e l'inviolabilità del sistema crittografico RSA (lez. 7-8).

Metodologie didattiche

Abbiamo strutturato il laboratorio come una sequenza di sistemi crittografici (e teorie matematiche ad essi connesse) sempre più evoluti (cioè matematicamente sofisticati) e difficili da violare.
Il passo da un sistema crittografico all'altro sarà guidato ma sarà, soprattutto, originato da un dibattito in classe: una volta forniti tutti gli strumenti per l'evoluzione, cercheremo di far nascere dagli studenti "l'idea giusta" gratificando, così, il loro "senso matematico" e coinvolgendoli attivamente.
Il tutto sarà coadiuvato da tavole con esercizi e da strumenti da noi realizzati.
Le schede di lavoro saranno varie: alcune di carattere prettamente esercitativo, tipo risoluzione di congruenze o di equazioni alle congruenze, per familiarizzare con le nuove tecniche; altre a carattere più crittografico, ad esempio cimentarsi con la cifratura e la decifratura di un testo, che, comunque, prevede l'utilizzo delle congruenze e del sistema binario, ma in modo più velato.
Gli strumenti che proporremo sono un modellino del disco cifrante di Leon Battista Alberti, che permette una verifica diretta e visiva del significato di congruenza e un programma al computer, implementato in C, che consente di generare velocemente i numeri primi.
Si potrebbe, infine, realizzare e presentare un altro programma che mostri la difficoltà di calcolo della fattorizzazione di un numero: sarebbe interessante avere la prova tangibile della differenza dei tempi che impiega un calcolatore nel generare un primo e nel fattorizzare un numero.

LEZIONE INTRODUTTIVA

La crittografia

La crittografia nasce fin dall'antichità dall'esigenza di possedere metodi efficienti per comunicare in modo segreto e sicuro, avendo la possibilità di inviare messaggi che possano essere letti rapidamente dai destinatari e non decifrati dal nemico, o da chiunque non sia autorizzato.
Il problema è estremamente attuale, con lo sviluppo dei sistemi elettronici che facilitano le comunicazioni, ma che le rendono anche molto vulnerabili se non vengono adeguatamente protette.
Per migliaia di anni re, regine e generali hanno avuto bisogno di comunicazioni efficienti e segrete per governare i loro paesi e comandare i loro eserciti: se informazioni preziose fossero cadute in mani ostili avrebbero potuto esserci conseguenze molto pericolose.
Per trovare il primo esempio di occultazione di messaggi possiamo risalire fino al V secolo a.C. alle guerre persiane.
Nelle *Storie* di Erodoto si legge che Serse, succeduto al padre Dario, dopo aver domato una ribellione in Egitto, si apprestava a muovere contro la Grecia.
Demarato, che aveva saputo dell'imminente spedizione, voleva avvertire gli Spartani senza correre il pericolo di essere scoperto dai nemici; ideò quindi uno stratagemma: prese una tavoletta doppia per scrivere, ne raschiò la cera e poi sul legno della tavoletta scrisse il piano di Serse. Fatto ciò, versò di nuovo la cera liquefatta sullo scritto in modo che, venendo portata vuota, la tavoletta non attirasse i sospetti da parte dei custodi delle strade. Ma quando la tavoletta giunse a Sparta, gli spartani non capirono cosa significasse fin quando la moglie di Leonida lo comprese e lo suggerì loro, invitandoli a raschiare la cera. Dandole ascolto trovarono il messaggio e lo spedirono agli altri greci che si prepararono così all'arrivo di Serse. Fu così che sconfissero i persiani nel 480 a.C.
Un messaggio segreto, abilmente nascosto, cambiò quindi le sorti di una guerra.

Un altro episodio, sempre dalle *Storie,* racconta che Istieo voleva dare ad Aristagora l'ordine di ribellarsi, ma essendo le strade sorvegliate non aveva modo di comunicarglielo.
Allora fece rasare la testa al suo schiavo più fidato, vi impresse dei segni e aspettò che gli ricrescessero i capelli. Non appena ricrebbero, lo spedì a Mileto con l'ordine, una volta giunto da Aristagora, di farsi radere i capelli. In questo caso però il trucco non ebbe successo, perché i nemici sospettarono che lo schiavo portasse qualche messaggio nascosto e quindi lo perquisirono in tutto il corpo, fino a rasargli la testa. Il messaggio, quindi, non fu mai consegnato al destinatario.

L'insieme dei trucchi come quelli usati in questi due esempi prendono il nome di *steganografia,* ossia procedura che consiste nel nascondere i messaggi, ed è la più antica e la più naturale forma di occultamento.
Ma questi due episodi appena raccontati sono sufficienti a mostrare la scarsissima sicurezza di questi sistemi: a chiunque sospetti che qualcuno possa aver nascosto un messaggio basterà ispezionare con la massima cura tutti i possibili nascondigli per scovarlo.

Perciò in parallelo con lo sviluppo della steganografia si assistè all'evoluzione della *crittografia*, dal greco *kryptòs,* che significa nascosto. La crittografia non mira a nascondere il messaggio in sé, ma il suo significato. Per rendere incomprensibile un testo, lo si altera per mezzo di un procedimento concordato a suo tempo dal mittente e dal destinatario. Quest'ultimo può invertire il procedimento e ricavare il testo originale.
Anche se crittografia e steganografia sono due discipline separate e indipendenti, possono in qualche modo essere considerate l'una come l'evoluzione dell'altra in termini di sicurezza.

La crittografia ha infatti il vantaggio di rendere il messaggio irriconoscibile e inutilizzabile dal nemico anche quando questi riesce ad intercettarlo.
Egli, infatti, può sospettare l'algoritmo utilizzato per cifrare, ma non la chiave. *????*

Uno dei primi esempi famosi di crittografia si trova nel *De Bello Gallico* di Cesare. L'autore racconta del riuscito invio di un messaggio a Cicerone, assediato e sul punto di arrendersi. Cesare usò una cifratura detta per *trasposizione,* che consisteva in un alfabeto *in chiaro* (quello ordinario) e un alfabeto *cifrante* ottenuto sostituendo ogni lettera dell'alfabeto ordinario con una lettera che lo rimpiazza nel crittogramma. Ad esempio

alfabeto chiaro	a b c d e f g h i l m n o p q r s t u v z
alfabeto cifrante	D E F G H I L M N O P Q R S T U V Z A B C

ci permette di scrivere le parole GIULIO CESARE come LNAONR FHVDUH.

In questo caso appare evidente che l'alfabeto cifrante non è altro che quello in chiaro traslato a destra di tre posizioni.
Questa particolare cifratura per trasposizione prende il nome di *cifratura di Cesare,* che studieremo in dettaglio e da cui partirà proprio il nostro laboratorio.

E' chiaro, però, che questo primo sistema appare facilmente attaccabile: le lettere dell'alfabeto italiano sono 21 e quindi possiamo traslarle al massimo di 20 posizioni: traslando di 21, infatti, torneremmo all'alfabeto chiaro. Quindi, anche se il nostro ipotetico nemico non conosce la chiave (ovvero non sa di quanti posti sono state traslate le lettere), sospettando che la cifratura sia di Cesare, avrebbe bisogno al massimo di 20 tentativi per risalire al messaggio originale!

E' per questo che i *crittologi* hanno avvertito da subito la necessità di studiare sistemi più sofisticati e sicuri. Anche perché, accanto alla nascita della *crittologia* (la scienza praticata dai crittologi che ha lo scopo di ideare nuove tecniche) si registra quella della *crittoanalisi,* cioè la scienza dell'interpretazione di un messaggio di cui si ignora la chiave. Mentre i crittologi, da un lato, mettono a punto nuovi sistemi di scrittura segreta, dall'altro i *crittoanalisti* cercano di individuare i loro punti deboli e carpirne i segreti.

Nessun esempio mostra le potenziali conseguenze della crittoanalisi in modo più drammatico del processo a Maria Stuarda di Scozia, il cui esito dipese completamente dallo scontro tra i suoi cifratori e i decrittatori di sua cugina, Elisabetta I.
La regina di Scozia, imprigionata da Elisabetta nel 1568, rimase prigioniera per 18 anni. Nel 1586 fu organizzato un piano per liberarla e contemporaneamente uccidere la regina Elisabetta. I cospiratori ritenevano, però, che il loro piano avrebbe dovuto avere l'approvazione di Maria. Per fare ciò si servirono di messaggi nascosti (steganografia) e anche cifrati (crittografia) che venivano recapitati in prigione da un messaggero. Uno dei cospiratori fece però il doppio gioco ma fu fatale alla regina di Scozia e ai suoi sudditi l'errata convinzione della inattaccabilità del loro sistema crittografico. I crittoanalisti di Elisabetta permisero alla sovrana di smascherare il piano, evitare che la congiura fosse portata a termine e impossessarsi di prove schiaccianti sul coinvolgimento della cugina nella cospirazione. Maria Stuarda fu così condannata a morte.

Ma, per inciso, l'importanza della crittoanalisi va oltre la semplice sfida contro i crittologi. Pensiamo, ad esempio, alle iscrizioni antiche: ovviamente l'intendimento di chi scriveva non era, in genere, di criptarle, ma per noi rappresentano di fatto un messaggio cifrato, che dobbiamo decifrare. La decifrazione di una scrittura sconosciuta rappresenta qualcosa di magico, perché ci permette di conoscere una civiltà ormai morta, di farci un quadro storico dell'epoca a cui l'iscrizione risale, ecc.

Si pensi alla decifrazione dei geroglifici egiziani: c'è un famoso reperto archeologico, la *stele di Rosetta,* di basalto nero, rinvenuta nel 1799 nei pressi delle foci del Nilo e incisa nel 196 a.C. Si tratta di un'iscrizione riguardante il decreto di un'assemblea di sacerdoti in onore del faraone ed è noto che essa reca lo stesso testo in tre versioni: egiziana geroglifica, egiziana demotica e greca. L'iscrizione in greco fu abilmente tradotta e diventò il "testo in chiaro" con cui confrontare le altre due scritture: essa costitutiva, quindi, un opportunità e una sfida irresistibile. J.F. Champollion raccolse la sfida e svelò nel 1822 il mistero dei geroglifici. La stele è oggi conservata nel British Museum a Londra.
Chiudiamo l'inciso con una curiosità: abbiamo detto poco fa che l'intendimento di chi incideva le iscrizioni non era, in genere, di criptarle. Abbiamo prudentemente scritto *in genere:* infatti alcuni scienziati hanno recentemente scoperto la presenza di procedimenti crittografici nei geroglifici egiziani. Sembra infatti che alcuni di essi siano stati crittati per ordine dei faraoni con varie tecniche, tra cui la cifratura per sostituzione.

La fine della cifratura per sostituzione monoalfabetica avvenne ad opera dei crittoanalisti arabi: furono loro i primi, non solo ad inventare in pratica la crittoanalisi, ma anche a escogitare una tecnica che permetteva di decifrare in breve tempo qualsiasi crittogramma: l'*analisi delle frequenze.*
In ogni lingua ci sono lettere che compaiono nei testi con maggiore frequenza ed altre più raramente, ad esempio nella lingua italiana le lettere più frequenti sono nell'ordine e,a,i mentre le meno usate sono q,z. Non solo: l'analisi del testo ci fornisce altre informazioni. Ad esempio nella lingua italiana la maggior parte delle parole termina con una delle vocali a,e,i,o e ciò vorrà dire che le lettere finali delle parole del messaggio cifrato dovranno essere una di queste lettere. Si può inoltre ricorrere ad altre considerazioni, come il fatto che lettere consecutive identiche necessariamente devono essere consonanti.
E questo strumento ha permesso (anche se con un po' più di tempo e fatica) ai crittoanalisti negli anni a seguire di distruggere la sicurezza di sistemi che via via i crittologi avevano complicato, come quelli che utilizzano parole come chiave oppure quelli polialfabetici, basati sul principio di cambiare alfabeto cifrante per ogni lettera del testo chiaro, come il cifrario di Vigenére che per anni è stato definito "il cifrario indecifrabile".

Le macchine per cifrare

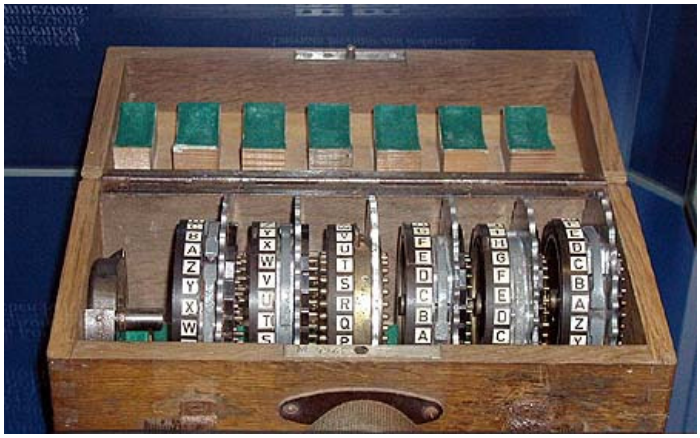
Col passare degli anni e col progresso della tecnologia, la cifratura divenne sempre più automatizzata.
Tuttavia il primo vero "macchinario" per cifrare (che era solo un supporto per la cifratura manuale) è il cosiddetto *disco cifrante* di Leon Battista Alberti. Esso consiste in due dischi di rame concentrici, di diametri differenti, che sono imperniati nel centro su di un perno e possono così ruotare l'uno rispetto all'altro attorno ad esso. Lungo le circonferenze dei due dischi sono incisi due alfabeti. Per crittare un messaggio con la cifratura di Cesare traslando di due lettere, basta collocare la *a* del disco interno (che rappresenta l'alfabeto in chiaro) in corrispondenza della *C* del disco esterno (che rappresenta ovviamente l'alfabeto cifrante). Una volta fatta questa operazione, per cifrare il messaggio basta di volta in volta, senza più ruotare i dischi, vedere quali sono le lettere sul disco esterno che corrispondono alle varie lettere del disco interno. Si tratta di un congegno molto semplice e molto efficace che è stato utilizzato per secoli. Questo stesso congegno può essere usato anche per una cifratura polialfabetica: basta, infatti, ruotare il disco di quante posizioni vogliamo per ogni lettera del messaggio.

Come detto, questa macchina ha resistito parecchi secoli, fino a quando, al termine della prima guerra mondiale, non è stata soppiantata da un nuovo macchinario tedesco, basato sempre su sistemi polialfabetici, ma in modo assai più ingegnoso.

Questa nuova macchina, chiamata *Enigma,* fu ideata e realizzata da Arthur Scheribius e Richard Ritter. Essa, almeno inizialmente, consisteva di tre elementi collegati da fili elettrici:

- una tastiera sulla quale erano disposte le lettere dell'alfabeto ordinario per poter battere il testo chiaro;
- uno *scambiatore,* che era il marchingegno che permetteva la cifratura vera e propria;
- un *visore*, sul quale erano disposte tante lampadine quante le lettere dell'alfabeto, in modo che l'impulso elettrico, dopo essere stato elaborato, andasse ad illuminare la lampadina corrispondente alla lettera crittata.

Lo scambiatore è un meccanismo costituito da uno spesso disco di gomma attraversato da una complessa rete di fili elettrici. Se ad esempio vogliamo criptare la lettera *a* con la lettere *D* (cioè lo scambiatore trasla di tre lettere), si digita *a* sulla tastiera: ciò facendo la corrente elettrica entra nello scambiatore seguendo il percorso dei fili, illuminando così la lampadina corrispondente alla lettera *D.* Gli scambiatori potevano essere sostituiti in base alla chiave che si voleva usare. Successivamente la macchina fu perfezionata e lo scambiatore divenne ruotante: questo faceva sì che il disco dopo la cifratura di ogni lettera ruotasse di un ventunesimo di giro in modo da usare un nuovo alfabeto per crittare ogni lettera. Lo scambiatore individuava, infatti, 21 alfabeti cifranti diversi.



In seguito Enigma divenne ancora più sofisticata, con l'introduzione di più dischi scambiatori e di *riflettori* in grado di riflettere i segnali elettrici e rendere ancora più complicata la cifratura.
Insomma, si trattava di una macchina molto efficace per cifrare, tanto che Scheribius era convinto che Enigma generasse dei messaggi inviolabili. Tuttavia la macchina, pur avendo le dimensioni di una ventiquattr'ore, pesava ben 12 kg e inoltre non esisteva possibilità di stampa, dunque l'operatore doveva copiare a mano, carattere per carattere il messaggio cifrato da trasmettere.
Per decodificare il crittogramma bastava possedere un'altra macchina Enigma e un cifrario con l'assetto degli scambiatori invertito. Tale assetto veniva cambiato giorno per giorno.
Anche durante la seconda guerra mondiale i tedeschi continuarono ad usare una versione portatile a batteria di Enigma. Infatti i crittoanalisti si trovarono di fronte alla difficoltà di avere troppi messaggi da decifrare e non abbastanza tempo.
Nel 1943, nel corso della seconda guerra mondiale, fu impiegato dagli Inglesi il sistema *Colossus* per decifrare i codici prodotti dal sistema Enigma.
Ma fu il famoso matematico Alan Turing all'inizio della guerra a violare la sicurezza di Enigma.

La rivoluzione della chiave pubblica: RSA

Nei cifrari fin qui descritti il processo di decifratura non presenta in generale grandi difficoltà una volta che sia noto il metodo usato per cifrare e cioè la chiave. Infatti in questi casi la funzione di decifratura è, per così dire, simmetrica a quella di cifratura, ossia è, sia da un punto di vista logico che computazionale, una funzione dello stesso tipo. In particolare tutti i crittosistemi classici si riferiscono allo scambio di messaggi tra due soli utenti e sono basati sulla condivisione di una chiave che consente, in sostanza, cifratura e decifratura.
Ma facciamo un passo avanti. In un'epoca come quella attuale in cui la maggior parte dell'informazione avviene via telefono o posta elettronica o radio, ogni messaggio inviato, come anche ogni trasmissione di chiave, è soggetta ad una facile intercettazione. E' pertanto indispensabile trovare nuovi modi più sicuri di comunicazione protetta.
Questo nuovo modo è costituito dalla crittografia a chiave pubblica.
Un cifrario a chiave pubblica è un sistema che permette di divulgare il metodo ed anche la chiave di cifratura (da cui il nome), senza per questo rivelare contestualmente il modo di decifrare. In altre parole, in tali sistemi, per calcolare in un tempo ragionevolmente breve la trasformazione di decifratura, inversa di quella di cifratura, è necessario essere in possesso di altre informazioni oltre quelle rese pubbliche e di cui abbiamo parlato. Tali informazioni però sono tenute segrete e senza di esse la complessità del calcolo della decifratura è tale da renderla implausibile.
Ed è in questo modo che funziona il sistema RSA (come vedremo alla fine del laboratorio), per mezzo di una funzione che possiamo definire "a senso unico".
Ciò che rende questo sistema "inattaccabile" è il fatto che i crittologi sono in vantaggio al momento sui crittoanalisti: questi ultimi, infatti, non sono ancora riusciti a trovare i mezzi per distruggere l'RSA. Ma ciò non vuol dire che sia davvero indecifrabile: nessuno può garantire, infatti, che in futuro non siano scoperti nuovi strumenti matematici in grado di superare anche questo nuovo sistema. E nessuno, inoltre, può essere certo che nuove scoperte non siano già state fatte. Enti come il National Security Agency (NSA) americano continuano a compiere studi ultrariservati sulla crittografia; ovviamente, le conoscenze che ne derivano non sono divulgate e i loro autori restano anonimi.

Numeri primi

Che cos'è un *numero primo*? La risposta appare semplice a primo impatto: sappiamo, infatti, già dalle scuole medie che un numero si dice *primo* quando non è divisibile per nessun altro numero se non per se stesso e per 1.
Apparentemente quindi questi numeri, pur avendo una caratteristica che li distingue dagli altri, non sembrano essere poi così importanti. Sono numeri con una particolare proprietà, così come ce ne sono tanti con proprietà diverse: ad esempio 4 ha la proprietà di essere pari oppure di essere il quadrato di 2; 6 ha la proprietà di essere un numero perfetto, cioè di essere la somma di tutti e soli i suoi divisori.
Ma guardiamo la proprietà dei numeri primi sotto un'altra luce: un numero primo, per com'è definito, non può essere scritto come prodotto di due numeri più piccoli (ovviamente diversi da se stesso e da 1), mentre qualsiasi altro numero possiede questa caratteristica: 35 non è primo e può essere scritto come 5x7, oppure 42 non è primo e si può scrivere come 6x7…ma iterando il ragionamento si osserva subito che in quest'ultima *fattorizzazione* lo stesso 6 non è primo, ma è dato dal prodotto di 2 e 3 e quindi possiamo riscrivere 42 come 2x3x7.
In entrambi i casi (e lo stesso vale per qualsiasi altro numero non primo) abbiamo scomposto il numero di partenza nel prodotto di numeri più piccoli *primi.*
Già da questa osservazione viene forse naturale il sospetto che questi numeri primi abbiano una caratteristica davvero speciale rispetto a quelli non primi: ci permettono di "generare" qualsiasi altro numero, semplicemente moltiplicandoli tra di loro. Per usare una metafora, è come se i primi

costituissero gli atomi della matematica, come se fossero una sorta di tavola periodica degli elementi (come quella usata in chimica che ci permette di creare le molecole presenti nel mondo fisico) a partire dai quali si "creano" i numeri.
E' chiaro che per un matematico la loro importanza non si esaurisce qui: esiste in matematica tutta una branca, che va sotto il nome di *teoria dei numeri*, in cui i primi assumono un ruolo centrale e che è continuamente oggetto di studio e di ricerca.
Non ci occuperemo di teoria dei numeri vera e propria in questo laboratorio, ma vedremo come la matematica e i numeri primi in particolare entrino nella vita pratica di tutti i giorni attraverso un potente strumento come il computer. Di come, cioè, i *crittologi* si servano di questa importante teoria per ideare e sviluppare dei sistemi in grado di *cifrare* messaggi in modo sicuro e di garantire la privacy di quanti vogliono comunicare privatamente a distanza, ma soprattutto la sicurezza di transazioni finanziarie attraverso le banche, i bancomat, le carte di credito ecc. Ogni volta che ordiniamo qualcosa su un sito web, il nostro computer usa la sicurezza fornita dall'esistenza di numeri primi di cento cifre, attraverso un sistema noto col nome di *RSA*.
Ad oggi sono più di un milione i numeri primi che sono già stati usati per proteggere il mondo del commercio elettronico.
Quindi la ricerca di numeri primi può apparentemente sembrare un'attività inutile. E fino a non molto tempo fa la reale importanza "pratica" di tale ricerca era nascosta perfino agli stessi matematici. G.H. Hardy, matematico di Cambridge, nel suo libro "Apologia di un matematico", riferendosi alla teoria dei numeri afferma:
<< Tanto un Gauss quanto dei matematici meno importanti possono a buona ragione rallegrarsi del fatto che qui c'è comunque una scienza la cui stessa lontananza dalle ordinarie attività umane dovrebbe mantenere amabile e pura>>.
Ma a partire dagli anni Settanta, come vedremo, i numeri primi conquistarono il centro della scena nel mondo del commercio.

Un concetto molto antico

I numeri primi erano noti fin dall'antichità, o almeno ci sono delle tracce e degli studi che ci permettono di ipotizzare che già molto prima della nascita di Cristo antiche civiltà come i greci e i cinesi fossero a conoscenza dell'esistenza di numeri "speciali", pur non comprendendone né apprezzandone a fondo la vera importanza.
La più antica, anche se incerta, prova è un osso risalente al 6500 a.C., l'osso di Ishango, scoperto nel 1950 fra i monti dell'Africa equatoriale centrale. Vi sono incise quattro serie di tacche disposte su tre file. In una delle file si contano 11, 13, 17 e 19 tacche, cioè un elenco completo dei numeri primi compresi tra 10 e 20. Non è chiaro se quest'antico osso, conservato all'istituto reale di Scienze Naturali di Bruxelles, rappresenti davvero uno dei primi tentativi da parte dei nostri antenati di comprendere i numeri primi oppure se le incisioni siano una scelta casuale di numeri e che solo per caso siano primi.
Ci sono anche prove del fatto che prima del 1000 a.C. i cinesi avevano elaborato un modo fisico per comprendere che cos'è che rende i primi, fra tutti i numeri, particolari: se prendete 15 fagioli, li potete disporre in un perfetto rettangolo composto da tre file di cinque righe; se però prendete 17 fagioli, l'unico rettangolo che potete costruire è quello formato da una sola riga di 17 fagioli.
Ma i primi a scoprire la reale potenza dei numeri primi furono i greci, nel IV secolo a.C.: essi compresero che ogni numero poteva essere creato moltiplicando fra loro dei numeri primi.
Per secoli quasi tutti i più grandi matematici si dedicarono allo studio dei primi, facendo molte scoperte importanti, ma ancora oggi il problema è aperto. Nessuno, infatti, è mai riuscito a stabilire il criterio con cui questi numeri si dispongono e a trovare un modo rapido per calcolarli. Non è infatti possibile sapere quale sarà, ad esempio, il millesimo numero primo senza calcolare i 999

precedenti. Oggi, in realtà, ciò è possibile poiché esistono delle tavole con i primi, che arrivano anche a numeri molto elevati, ma sono state compilate con grossi sforzi e nel corso di centinaia di anni.
La prima tavola risale a Eratostene, bibliotecario ad Alessandria, che nel III secolo a.C. scoprì una procedura ragionevole per determinare quali fossero i numeri primi compresi, per esempio, tra 1 e 1.000, che va sotto il nome di *Crivello di Eratostene.* Egli scriveva per esteso i numeri da 1 a 1.000, poi prendeva il numero primo più piccolo, cioè 2, e a partire da quello depennava dall'elenco un numero ogni due, ovvero i multipli di 2. I numeri cancellati non erano primi perché erano divisibili per 2. Quindi passava al numero successivo che non era stato eliminato, ovvero3, e a partire da quello depennava tutti i multipli di 3 che, essendo divisibili per 3, non erano primi. E continuava così, prendendo il numero successivo che non era stato cancellato e depennando tutti i suoi multipli: ogni numero primo crea un "crivello" che permette di eliminare una parte dei numeri non primi. Arrivato a 1.000, i numeri rimasti sono tutti primi.
La cosa che saltò all'occhio di tutti i matematici fu che in queste tavole non c'era un ordine apparente, che i numeri primi sembravano essere disposti totalmente a caso. Non c'è una formula che li lega, non c'è un criterio per stabilire, dato un qualsiasi primo, quale sarà quello successivo senza andare più o meno per tentativi su tutti i numeri.
Questo problema ha accompagnato e accompagna tuttora i matematici di tutto il mondo: la *struttura* che si nasconde dietro la sequenza dei primi. Tutti noi almeno una volta ci siamo cimentati nella risoluzione di giochi in cui data una sequenza di numeri si deve trovare il successivo ricostruendo la regola che c'è dietro, come ad esempio:
1, 1, 2, 3, 5, 8, 13,…. in cui ciascun numero è somma dei due precedenti (nota come la sequenza di *Fibonacci*), ma una regola simile per i numeri primi non è stata ancora trovata. Esiste perfino un cospicuo premio in denaro per chi riesca a risolvere questa *congettura.*
Molti nomi illustri hanno finora fallito: Euclide, Fermat, Eulero, Gauss, Riemann e molti altri ancora. Il loro contributo è stato indubbiamente fondamentale: Euclide negli *Elementi* ha dimostrato che esistono infiniti numeri primi; Fermat pensava di aver trovato una formula che permettesse di trovare alcuni numeri primi (anche se non tutti) e di fornirne almeno un elenco in modo piuttosto semplice:
si trattava dei numeri della forma $2^{2^N}+1$, ma in seguito si scoprì che Fermat sbagliava perché per *N=5* non si ottiene un numero primo, solo che gli strumenti a sua disposizione erano insufficienti per scoprirlo.
Sulla sua scia si mise Mersenne, un monaco francese appassionato di matematica, che sistemando ciò che aveva fatto Fermat scoprì che i numeri della forma 2^N-1 per alcuni valori di *N* davano effettivamente un numero primo. Con gli scarsi mezzi che aveva a disposizione riuscì misteriosamente a scoprire che era primo il numero con *N=257*, un numero di ben 77 cifre! Ma ancora oggi non sappiamo se gli *N* per cui si ottiene un primo siano infiniti. Il più grande primo di Mersenne conosciuto finora è il 37esimo, un numero a ben 909.526 cifre!

Eulero, invece, realizzò una tavola dei primi fino a 100.000, ma una delle sue scoperte più curiose fu una formula che si serviva dei polinomi per generale una quantità inspiegabile di primi, ma non riuscì a perfezionarla.
Ad esempio, inserendo (come egli stesso fece) tutti i numeri compresi tra 0 e 39 nel polinomio x^2+x+41, ottenne il seguente elenco di primi:

41, 43, 47, 53, 61, 71, 83, 97, 113, 131, 151, 173, 197, 223, 251, 281, 313, 347, 383, 421, 461, 503, 547, 593, 641, 691, 743, 797, 853, 911, 971, 1.033, 1.097, 1.163, 1.231, 1.301, 1.373, 1.447, 1.523, 1.601.

E lo stesso lungo elenco poteva essere costruito usando *q*=2, 3, 5, 11, 17 nella formula al posto di 41 e inserendo tutti i numeri tra 0 e *q-2.*

Ma nonostante questa scoperta, lo stesso Eulero disse: <<Ci sono alcuni misteri che la mente umana non penetrerà mai. Per convincercene non dobbiamo far altro che gettare un'occhiata alle tavole di numeri primi. Ci accorgeremo che non vi regna né ordine né legge>>.

Fra tutti Gauss e Riemann fecero, forse, le più grandi scoperte.
Il primo riuscì a stimare il numero di primi presenti in un intervallo, attraverso una funzione che usava il logaritmo e che indicava con $\pi(N)$.
Da questo calcolo cominciava ad emergere una struttura: per esempio, il numero di primi compresi tra 1 e 100 è 25. Gauss, servendosi delle tavole di Euclide che aveva a disposizione, osservò come cambiava questa funzione tra 1 e 1.000 e tra 1 e 10.000 e così via e realizzò una tabella simile a quella riportata qui sotto, in cui inserì anche la distanza media tra due primi successivi:

N	Numero di primi compresi fra 1 e *N*, che spesso si indica con $\pi(N)$	Distanza media fra due numeri primi successivi
10	4	2,5
100	25	4,0
1.000	168	6,0
10.000	1.229	8,1
100.000	9.592	10,4
1.000.000	78.498	12,7
10.000.000	664.579	15,0
100.000.000	5.761.455	17,4
1.000.000.000	50.847.534	19,7
10.000.000.000	455.052.511	22,0

Nell'ultima colonna a destra emerge questa possibile struttura: se ad esempio consideriamo l'intervallo da 1 a 100 (prima colonna) abbiamo 25 numeri primi (seconda colonna) e quindi uno ogni 4 (terza colonna), quindi in media per trovare il numero primo successivo bisogna aggiungere 4. Se consideriamo i numeri fino a 10.000, troviamo 1.229 primi, in media uno ogni 8.1. E si può osservare, inoltre, che superato 10.000 l'incremento della terza colonna è sempre di circa 2.3.
Perciò Gauss scoprì che ogni volta che moltiplicava *N* per 10 doveva aggiungere 2.3 circa al rapporto fra i numeri primi e *N* . E semplicemente osservando che tale relazione tra moltiplicazione e addizione è la stessa che intercorre tra i logaritmi (quando si moltiplicano due numeri i loro logaritmi si sommano) Gauss riuscì a trovare un nesso importante tra numeri primi e logaritmi in base *e*, ovvero che per i numeri compresi fra 1 e *N,* grossomodo ogni *log N* numeri ce ne sarà uno primo.

[altre osservazioni sulla tabella potrebbero farle i ragazzi, tipo che il rapporto ci da la probabilità di trovare un primo nell'intervallo (se lo sanno) oppure che il rapporto aumentando N diminuisce e quindi la probabilità di trovare i primi è di meno].

E, a giudizio di Gauss stesso, aumentando *N* questa approssimazione risultava sempre migliore, ma non fu in grado di dimostrarlo.

Ci riuscì parecchi anni dopo Bernhard Riemann, che non solo perfezionò la funzione ideata da Gauss e la trasformò in un'altra funzione *R(N)* che approssimava con maggior precisione il numero dei primi presenti in un certo intervallo, come si può apprezzare dalla tabella

N	Numero di primi $\pi(N)$ compresi fra 1 e N	Sovrastima data dalla funzione di Riemann *R(N)*	Sovrastima data dalla funzione di Gauss Li(*N*)
10^2	25	1	5
10^3	168	0	10
10^4	1.229	-2	17
10^5	9.592	-5	38
10^6	78.498	29	130
10^7	664.579	88	339
10^8	5.761.455	97	754
10^9	50.847.534	-79	1.701
10^{10}	455.052.511	-1.828	3.104
10^{11}	4.118.054.813	-2.318	11.588
10^{12}	37.607.912.018	-1.476	38.263
10^{13}	346.065.536.839	-5.773	108.971
10^{14}	3.204.941.750.802	-19.200	314.890
10^{15}	29.844.570.422.669	73.218	1.052.619
10^{16}	279.238.341.033.925	327.052	3.214.632

ma diede vita a una delle teorie più celebri della matematica dall'800 ad oggi: la *congettura di Riemann.*

Si tratta di una teoria abbastanza sofisticata (su cui non vogliamo intrattenerci) che servendosi di funzioni sinusoidali e numeri immaginari riesce a ipotizzare a ragione un ordine nell'apparente casualità dei numeri primi. Riemann ebbe il grande merito di "guardare" le cose nel modo giusto: l'ordine c'è sempre stato, i mezzi per scoprirlo anche…era necessario servirsene nel modo giusto.

Tuttavia Riemann non riuscì a dimostrare la sua congettura, la questione è attualmente aperta e c'è perfino un premio in denaro di un milione di dollari per chiunque riesca a dimostrarla!

Congruenze

Un altro potente strumento matematico di cui si serve la crittografia è il concetto di congruenza, che impareremo ad usare in questo laboratorio.
La prima intuizione su questo argomento è dovuta ancora una volta a Gauss, con l'invenzione del suo *calcolatore a orologio.*
Il nome è dovuto al fatto che l'idea è analoga al funzionamento di un orologio, anche se nella mente di Gauss non si trattava di una macchina materiale, ma solo di un'idea astratta: se l'orologio dice che sono le 9 e aggiungiamo 4 ore, la lancetta delle ore si sposta sull'una. Allo stesso modo il calcolatore a orologio di Gauss fornisce 1 invece di 13 come risultato di 9+4. E volendo fare un calcolo più "complicato", come ad esempio 7x7, il calcolatore a orologio gli forniva il risultato della divisione di 49 (ossia 7x7) per 12, cioè di nuovo 1.
Ma era quando Gauss voleva calcolare 7x7x7 che la potenza dello strumento cominciava ad emergere: invece di moltiplicare un'altra volta 49x7, Gauss poteva limitarsi a moltiplicare per 7 l'ultimo risultato ottenuto, cioè 1, per ottenere la risposta, cioè 7, senza dover calcolare effettivamente 7x7x7=343. Egli sapeva così con minor fatica quanto faceva 343 diviso 12.
Ma il calcolatore dimostrò davvero tutta la sua potenza quando Gauss cominciò ad usarlo con numeri molto grandi, che oltrepassavano le sue stesse capacità di calcolo manuale. Ad esempio, pur non avendo idea di quanto facesse 7^{99}, il suo calcolatore a orologio col procedimento appena descritto gli diceva che quel numero diviso per 12 dava come resto 7.
Il passo successivo avvenne quando Gauss si rese conto che non c'era nulla di speciale nel numero 12 (quello utilizzato dall'orologio) e che quindi l'idea poteva essere applicata a qualsiasi numero intero. Introdusse, perciò, l'idea di una nuova aritmetica, chiamata *aritmetica modulare,* basata su orologi con qualsiasi numero di ore. Per esempio, se inseriamo il numero 11 in un calcolatore a orologio diviso in 4 ore, la risposta che otterremo sarà 3, come il resto della divisione di 11 per 4.
Impareremo quindi ad usare questa nuova aritmetica e capiremo come il mondo della crittografia e della sicurezza informatica si basano su questa geniale idea di Gauss. Anche se si usano in questo ambito immaginari "orologi" con numeri di ore dell'ordine di centinaia di cifre.

Matematica e crittografia

Come si legano tra loro i due concetti cardine del nostro laboratorio, le congruenze e i numeri primi, con la crittografia? E' questo l'obiettivo che ci prefiggiamo di raggiungere al termine di questo percorso laboratoriale; scopriremo, infatti, che le funzioni di cifratura e decifratura (di tutti i sistemi che passeremo in rassegna) si tengono in piedi sulla solida struttura delle congruenze e che la sicurezza attuale del sistema RSA, cui abbiamo accennato precedentemente, si basa quasi esclusivamente sulla nostra incompleta conoscenza dei numeri primi. Daremo a tal proposito nella lezione conclusiva un'idea della *fattorizzazione* dei numeri in primi e vedremo quali difficoltà presenta sia manualmente che con l'importante contributo del calcolatore.

Prima lezione

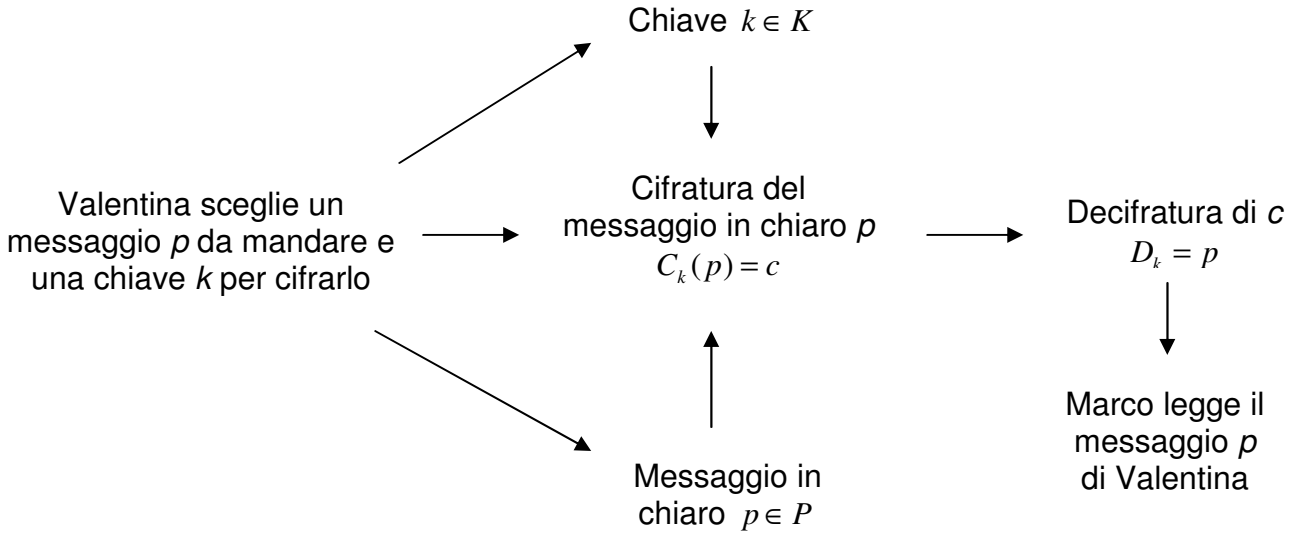
CIFRARIO DI CESARE

Il nostro intento è analizzare in dettaglio i metodi di cifratura che si sono susseguiti nel corso della storia prestando particolare attenzione all'impianto matematico che ne consente la realizzazione.
Iniziamo il nostro studio approfondendo il primo esempio di crittografia che la storia ci fornisce. Svetonio, uno storico del II sec d.C., nella sua Vita dei Cesari parla di un sistema utilizzato da Cesare per cifrare i suoi messaggi: egli spostava di tre lettere ogni lettera del messaggio da inviare.

Innanzi tutto ricordiamo che un crittosistema è costituito da:

- l'insieme dei messaggi in chiaro *P* di cui indichiamo gli elementi con la lettera p;
- l'insieme delle chiavi *K* in cui ogni elemento k determina una trasformazione di cifratura C_k e una trasformazione di decifratura D_k che sono una l'inversa dell'altra;
- l'insieme dei messaggi cifrati *C* i cui elementi sono indicati con la lettera c.

Abbiamo quindi che un crittosistema è determinato da (*P,K,C*) e la comunicazione tra due persone, Valentina e Marco, può essere riassunta dal seguente diagramma:



Nel cifrario di Cesare:

- gli elementi p sono le parole che vogliamo inviare;
- la chiave consiste in fase di cifratura nello spostare di tre posti le varie lettere (C_k) e in fase di decifratura nel rimetterle nella loro corretta posizione (D_k);
- gli elementi c sono il risultato dell'operazione di cifratura.

Se indichiamo con lettere minuscole le 21 lettere dell'alfabeto, ciascuna lettera del nostro messaggio (testo in chiaro) sarà sostituita con la lettera che si trova tre posizioni più avanti, e che per comodità indicheremo con caratteri maiuscoli, ottenendo così un nuovo messaggio (testo cifrato) apparentemente privo di significato

a b c d e f g h i l m n o p q r s t u v z
D E F G H I L M N O P Q R S T U V Z A B C

Ad esempio se il messaggio da inviare è il seguente:
domani non andiamo a scuola

il risultato dopo la cifratura sarà:
GRPDQN QRQ DQGNDPR D VFAROD

Possiamo decidere di generalizzare questo sistema utilizzando una chiave diversa cioè decidendo di spostare le lettere non più di tre posizioni ma di una quantità arbitraria.
Un sistema di questo tipo, in cui l'alfabeto cifrante è ottenuto dall'alfabeto in chiaro spostando di un certo numero di posizioni le lettere, prende il nome di **cifrario di Cesare** o di **cifratura per traslazione**.
Le possibilità per i cifrari di Cesare nel caso della lingua italiana sono solamente 20 perché ovviamente se una lettera si sposta di 21 posizioni, ritorna al punto di partenza. Mentre nel caso dell'alfabeto inglese abbiamo 25 alfabeti cifranti possibili dato che le lettere sono 26.

Abbiamo visto che volendo cifrare un messaggio usando il metodo di Cesare dobbiamo sostanzialmente traslare le lettere di un certa quantità di posizioni (che decidiamo noi e rappresenta la chiave utilizzata per cifrare).
In effetti possiamo interpretare così la procedura che attuiamo: assegnamo ad ogni lettera dell'alfabeto in chiaro un numero corrispondente alla sua posizione (la A occupa la posizione 0 e la Z la posizione 20), dopodichè decidiamo un numero (ad esempio 5) che rappresenta la nostra chiave e lo sommiamo ad ogni posizione così da ottenere che nell'alfabeto cifrante la A corrisponda alla lettera in posizione 5 cioè alla F, la B alla G,…, la R alla Z, la S che occupa la posizione 16 corrisponda alla A, la T alla B,…., la Z alla E.
Il nodo fondamentale di questa procedura sta nel fatto che quando abbiamo deciso la corrispondenza tra la T e la B abbiamo sostanzialmente ragionato cosi:

- la lettera T occupa la posizione 17 (ricorda che partiamo dalla posizione 0),
- 17+5=22 ma le posizioni possibili sono 21 e sono numerate da 0 a 20 quindi il numero 22 non corrisponderebbe a nessuna lettera
- $22 = 1 \cdot 21 + 1$ e abbiamo deciso che la lettera T doveva corrispondere a quella in posizione 1 cioè alla B

Ed è per questo motivo che la lettera S corrisponde alla A (perché S= posizione 16, 16+5=21, $21 = 1 \cdot 21 + 0$ e la lettera A occupa la posizione 0), la U alla C e così via.
Quindi da un punto di vista matematico quando cifriamo con questo metodo operiamo una somma (per traslare) dopodichè se il risultato è maggiore di 21 ci interessiamo solo al resto della divisione.
Questo modo di procedere può sembrare astruso ma in realtà lo utilizziamo quotidianamente ad esempio quando leggiamo l'ora sull'orologio. Infatti quando leggiamo l'orologio sappiamo benissimo che, ad esempio, le 13:00 corrispondono all'1:00 e le 18:00 alle 6:00, e il motivo di questa corrispondenza sta nel fatto che il resto della divisione per 12 di 13 è 1 mentre di 18 è 6.

Cerchiamo di generalizzare il concetto.

Innanzi tutto diamo una definizione cercando di capire che connessione ha con quanto detto finora.

Definizione Sia *n* un intero positivo fissato. Due numeri $a, b \in Z$ sono ***congrui modulo n*** se e solo se *a-b* è un multiplo di *n*, ovvero, espresso in formule, $a \equiv b \quad (n) \Leftrightarrow (a-b) = n \cdot h$ per qualche $h \in Z$.

Esempi:

1. $25 \equiv 1 \ (3)$ perché $25-1=24=3\cdot\mathbf{8}$
2. $67 \equiv 55 \ (6)$ perché $67-55=12=6\cdot\mathbf{2}$
3. $55 \equiv 1 \ (6)$ perché $55-1=54=6\cdot\mathbf{9}$
4. $-5 \equiv 1 \ (6)$ perché $-5-1=6\cdot\mathbf{1}$

Osservazioni.

1. Notiamo che gli esempi 2 e 3 ci suggeriscono l'idea di transitività della congruenza. Infatti vale anche che $67 \equiv 1 \ (6)$ perché $67-1=66=6\cdot\mathbf{11.}$

2. Ora possiamo dire che quando dobbiamo criptare un messaggio operiamo modulo 21. Per di più invece di lavorare con tutti i possibili numeri interi lavoriamo solo con quelli compresi tra 0 e 20 (che sono le posizioni possibili). Infatti se, per assurdo, volessimo operare una traslazione di 63 posizioni, per trovare come criptare, ad esempio, la lettera B dovremmo capire a quale numero b è congruo modulo 21 il numero a=64=63+1 (che rappresenta la posizione traslata della lettera B). Ovviamente scoprire che $64 \equiv 43 \ (21)$ anche se è un risultato esatto non ci fornisce nessuna informazione utile perché la lettera numero 43 non esiste! Quindi, in realtà, quando vogliamo capire a "cosa" è congruo un certo numero a modulo n siamo interessati a trovare quell'unico numero b compreso tra 0 e n-1 tale che sia verificata la congruenza.

Mettendo insieme le due osservazioni precedenti possiamo ricavarne che modulo 6 i numeri 67, 55, -5, -23 sono in realtà lo stesso numero e cioè 1 (provare per credere!!!). Perché? Perché il resto della divisione per 6 di questi numeri è sempre congruo a 1 modulo 6. Infatti 67/6 dà resto 1 (perché $67=6\cdot 11+1$) e banalmente $1 \equiv 1 \ (6)$ oppure -23/6 dà per resto -5 (perché $-23=6\cdot(-3)-5$) e $-5 \equiv 1 \ (6)$, e così via.

Quindi, in generale, possiamo concludere che, fissato n, tutti i numeri 0,….,n-1 siano in un certo senso speciali in quanto un qualsiasi altro numero intero (positivo o negativo) sarà congruo ad uno di essi modulo n.
Questo ruolo speciale sta proprio nel fatto che scelto un qualsiasi b tale che $0 \le b \le n-1$ tutti gli altri numeri interi il cui resto della divisione per n è uguale a b possono essere identificati con b stesso nel senso della congruenza.

Restano così definiti n insiemi di numeri ovvero:

Definizione Gli insiemi della forma $\overline{b}=\{a \in Z \mid a \equiv b(n)\}=$ interi che divisi per n danno resto b si chiamano ***classi di resto modulo n***. I numeri b sono tali che $0 \le b \le n-1$ e vengono chiamati ***rappresentanti della classe***.

ESEMPIO Possiamo calcolare tutte le classi di resto modulo 4:
$\overline{0}=\{.....,-16,-12,-8,-4,0,4,8,12,....\}=$ interi che divisi per 4 danno resto 0
$\overline{1}=\{...,-11,-7,-3,1,5,9,....\}=$ interi che divisi per 4 danno resto 1
$\overline{2}=\{....,-6,-2,2,6,10,...\}=$ interi che divisi per 4 danno resto 2
$\overline{3}=\{...,-9,-5,-1,3,7,11,....\}=$ interi che divisi per 4 danno resto 3.

Definizione L'insieme di queste classi di numeri si indica con Z_n. Cioè $Z_n=\{\overline{0},\overline{1},...,\overline{n-1}\}$.

ESERCIZI

- Specificare tutti gli elementi di Z_{21}.
- A che classe di resto modulo 23 appartiene il numero 74?
- Dire a che classe di resto modulo 5 appartiene il numero -7.
- Indicare la classe di resto modulo 12 di -13.
- Specificare la classe di resto modulo 7 di 63.

Allora, riassumendo in una tabella l'associazione lettera-numero fatta precedentemente,

Lettera	Numero	Lettera	Numero	Lettera	Numero
a	0	h	7	q	14
b	1	i	8	r	15
c	2	l	9	s	16
d	3	m	10	t	17
e	4	n	11	u	18
f	5	o	12	v	19
g	6	p	13	z	20

possiamo notare che l'alfabeto numerico dei messaggi unitari (le singole lettere) è rappresentato da $P = Z_{21}$.
Poiché nel cifrario di Cesare ogni lettera viene sostituita con la lettera che si trova un certo numero di posizioni più avanti abbiamo che l'insieme delle chiavi è $K=\{0,1,...,20\}$ (perché?)
Allora il sistema crittografico per traslazione può essere cosi schematizzato:
data la chiave $k \in K$, la funzione cifrante sarà la seguente:

$$C_k : Z_{21} \to Z_{21}$$
$$p \to p + k \mod 21,$$

mentre la funzione inversa, quella di decifratura, sarà:

$$D_k : Z_{21} \to Z_{21}$$
$$c \to c - k \mod 21.$$

ESERCIZI DI CIFRATURA E DECIFRATURA. Utilizzando le congruenze

1. Criptare un messaggio a scelta utilizzando k=7
2. Decriptare il seguente messaggio sapendo che k=16
   DE UHMNH GPFZMD IMDFD Z UMDOOHBMSADS Z DGOZMZNNSGOZ
   (il corso numeri primi e crittografia è interessante)

DECIFRATURA

Se intercettiamo un messaggio che sappiamo essere stato criptato col metodo della traslazione per decifrarlo dovremo scoprire quanto vale *k*, cioè la chiave. In realtà chi ha qualche esperienza di risoluzione di problemi crittografici tipo quelli della settimana enigmistica sa che è possibile decodificare un messaggio pur non conoscendo la chiave *k*.
Nel caso del cifrario di Cesare è possibile pensare di procedere per tentativi. Infatti i cifrari possibili sono 20 nel caso di un testo scritto in lingua italiana (25 se il testo è in inglese). Ma se il testo è

molto lungo i tempi di decifratura diventerebbero lenti fornendo magari troppo tardi informazioni utili.
Messi di fronte ad un testo da decifrare si procede con quella che si chiama **analisi del testo**.
Poiché nella lingua italiana la maggior parte delle parole termina con una delle vocali *a,e,i,o* vorrà dire che le lettere finali delle parole del messaggio cifrato dovranno essere una di queste lettere. Purtroppo si fa presto ad eludere questa considerazione spezzando il messaggio in blocchi della stessa lunghezza il che rende complicata la ricostruzione delle singole parole. Però si può ricorrere ad altre considerazioni. Se nel testo ci sono lettere consecutive identiche queste necessariamente devono essere consonanti. Si hanno poi ulteriori informazioni, fornite dalla cosiddetta **analisi delle frequenze**.
In ogni lingua ci sono lettere che compaiono nei testi con maggiore frequenza ed altre più raramente, ad esempio nella lingua italiana le lettere più frequenti sono nell'ordine *e,a,i* mentre le meno usate sono *q,z*. Riportiamo di seguito una tabella riassuntiva.

Lettera	%	Lettera	%	Lettera	%
a	11,74	h	1,54	q	0,51
b	0,92	i	11,28	r	6,38
c	4,50	l	6,51	s	4,98
d	3,73	m	2,52	t	5,63
e	11,79	n	6,88	u	3,02
f	0,95	o	9,83	v	2,10
g	1,65	p	3,05	z	0,49

Altre informazioni si possono reperire dalla frequenza delle doppie, dalla tendenza di certe lettere a non gradire la vicinanza di altre, ecc.

Supponiamo di intercettare il seguente messaggio:
OD LDZZD IUHZZRORVD ID N LDZZNQN FNHFMN
e di voler scoprire cosa significhi.

Riportiamo la frequenza delle lettere nel nostro messaggio:

Lettera	Occorrenze	Lettere	Occorrenze	Lettera	Occorrenze
A	0	H	2	Q	1
B	0	I	2	R	2
C	0	L	2	S	0
D	6	M	1	T	0
E	0	N	5	U	1
F	2	O	2	V	1
G	0	P	0	Z	6

.
La lettera D potrebbe essere una a, una e o una i. Lo stesso dicasi per le lettere N e Z. Tuttavia la Z è quasi sicuramente una consonante doppia perché non si possono avere due vocali consecutive uguali e dunque necessariamente la D, la N, la R e la H sono vocali, perché compaiono in …ZZD…, …ZZN e …HZZR. Molto probabilmente la O è una l visto che abbiamo stabilito che D è una vocale mentre la Z potrebbe essere una r, una n o una t per via della sua alta frequenza.
È evidente che abbiamo notevolmente ridotto il numero dei tentativi da fare per decodificare il messaggio.
Tentiamo con la sostituzione:
N=i, D=a, O=l, H=o, R=u, Z=t,

otteniamo:
la Latta IUottuluVa Ia i LattiQi FioFMi.
È chiaro che la scelta R=u non è proprio felice e dunque prendendo R=o e H=e arriviamo a
la Latta IUettoloVa Ia i LattiQi FieFMi
e con altri tentativi possiamo giungere alla soluzione:
la gatta frettolosa fa i gattini ciechi

È chiaro che il modo di procedere è molto falsato perché stiamo facendo l'analisi di un testo troppo breve, ma l'importante è aver sottolineato come le tante possibilità teoriche possano essere notevolmente ridotte usando informazioni sul linguaggio e procedendo sistematicamente per tentativi.

Seconda lezione

ANCORA SUL CIFRARIO DI CESARE

La cifratura di messaggi ha un ruolo così importante che nel tempo si sono sviluppati macchinari per cifrare più velocemente.

Il primo esempio di macchina per cifrare è il cosiddetto disco cifrante di Leon Battista Alberti. Esso consiste di due dischi concentrici, di diametri differenti, che sono imperniati nel centro su di un perno, e possono così ruotare l'uno rispetto all'altro attorno ad esso: lungo le circonferenze dei due dischi sono incisi due alfabeti. Nella circonferenza interna c'è l'alfabeto in chiaro, in quella esterna l'alfabeto cifrante. Per criptare un messaggio con la cifratura per traslazione utilizzando come chiave k=5 basta collocare la lettera a del disco interno in corrispondenza della lettera F sul disco esterno. Fatta questa semplice operazione ad ogni lettera dell'alfabeto in chiaro resta associata una lettera dell'alfabeto cifrato e così senza dover più ruotare i dischi si può trascrivere il messaggio vedendo quale lettera del disco esterno corrisponde alla lettera del disco interno che dobbiamo scrivere.

Abbiamo visto che questo metodo, che opera semplicemente traslando, è facilmente attaccabile sia con l'analisi delle frequenze sia per tentativi, cioè provando tutte le sole 20 combinazioni possibili.
È necessari quindi trovare un metodo di cifratura più efficace e sicuro.
Abbiamo visto che considerando le lettere come elementi di Z_{21} potevamo modellizzare la procedura di cifratura per traslazione nel modo seguente:
data la chiave $k \in K$, la funzione cifrante è:

$$C_k : Z_{21} \to Z_{21}$$
$$p \to p + k \mod 21,$$

mentre la funzione inversa, quella di decifratura, è:

$$D_k : Z_{21} \to Z_{21}$$
$$c \to c - k \mod 21.$$

La traslazione offre poche possibilità (solo 20) perché è un procedimento troppo semplice: tre lettere consecutive dell'alfabeto in chiaro (ad esempio a, b, c) vengono cifrate con tre le lettere consecutive dell'alfabeto cifrante (ad esempio D,E,F se la chiave è k=3). Bisognerebbe eliminare questa regolarità, questo schema con cui si susseguono le lettere. Per farlo è necessario "complicare" la funzione di cifratura C_k. *Discussione*

L'idea è quella di mescolare le lettere e di disporle in un ordine apparentemente casuale ma che almeno per noi e per il destinatario del nostro messaggio mantenga una logica ben precisa.

PROPRIETA' DI Z_n

Dato che, ormai, stiamo lavorando nel nuovo insieme Z_{21} per capire meglio come procedere bisogna analizzare più in dettaglio le caratteristiche e le proprietà di questo insieme; trattiamo il caso generale Z_n delle classi di resto modulo n.

Proprio come in Z , vi si possono definire operazioni che ci consentono di trattare le classi si resto quasi come numeri interi. E' possibile definire somma, prodotto e creare tutta un'aritmetica che viene definita aritmetica modulare perché si lavora modulo n (nel senso delle congruenze).

DEF. Date $\overline{a}$ $\overline{b}$ due classi di resto modulo n si ha che:

- $\overline{a}+\overline{b}=\overline{a+b}$
- $\overline{a}\cdot\overline{b}=\overline{a\cdot b}$

Per comodità di calcolo esprimeremo il risultato di queste operazioni sempre tramite il suo rappresentante cioè se, ad esempio, stiamo lavorando modulo 5 e dobbiamo calcolare la somma tra $\overline{3}$ e $\overline{4}$ faremo $\overline{3}+\overline{4}=\overline{7}=\overline{2}$ poiché $7\equiv 2\ (5)$ e similmente per il prodotto.

Esercizi su somma e prodotto e potenze.

- Calcolare $\overline{19}+\overline{7}$ modulo 21
- Calcolare $\overline{4}\cdot\overline{7}$ modulo 21
- A che classe modulo 21 appartiene $5\cdot 9+12$

Osserviamo che la definizione di queste di queste operazioni è ben posta cioè è indipendente dal rappresentante della classe che si sceglie per operare.
Infatti:
$\overline{18}+\overline{21}=\overline{39}\ \ (4)$
e $\overline{39}=\overline{3}\ (4)$ perché $39=4\cdot 9+3$
ma $\overline{18}=\overline{2}$ (perché $18=4\cdot 4+2$) e $\overline{21}=\overline{1}$ perché $21=4\cdot 5+1$ allora
$\overline{18}+\overline{21}=\overline{2}+\overline{1}=\overline{3}=\overline{39}$
Lo stesso vale per il prodotto:
$\overline{5}\cdot\overline{4}=\overline{20}=\overline{2}\ \ (6)$
e $\overline{17}\cdot\overline{10}=\overline{170}=\overline{2}\ (6)$
perché $\overline{17}=\overline{5}\ (6)$ e $\overline{10}=\overline{4}\ (6)$

Per comodità di calcolo, come abbiamo specificato anche in fase di definizione, sceglieremo sempre come rappresentanti delle classi di resto modulo *n* i numeri $\overline{b}$ tali che $0\le b\le n-1$.

Inoltre notiamo che entrambe le operazioni definite sono dotate di un elemento particolare, proprio come in Z dove esistono lo 0 che non modifica una somma (ad es. 12934+0=12934) e l'1 che non cambia un prodotto (ad es. $23458\cdot 1=23458$). In Z_n essi sono rispettivamente $\overline{0}$ e $\overline{1}$ infatti preso un altro qualsiasi elemento $\overline{a}\in Z_n$ vale che:
$\overline{a}+\overline{0}=\overline{a}$
$\overline{a}\cdot\overline{1}=\overline{a}$

Questo deriva dalla definizione delle operazioni somma e prodotto. *Si può discuterne*
Infatti:
$\overline{a}+\overline{0}=\overline{a+0}=\overline{a}$
$\overline{a}\cdot\overline{1}=\overline{a\cdot 1}=\overline{a}$

Non bisogna però pensare che tutto quello che vale per l'uguaglianza valga automaticamente anche per la congruenza cioè non tutte le operazioni con cui siamo soliti lavorare in Z restano valide in Z_n. Ad esempio la legge di cancellazione $a\cdot b=a\cdot c\Rightarrow b=c$ che vale in Z purché sia $a\neq 0$ non si trasporta alle congruenze ad esempio:
$3\cdot 5\equiv 3\cdot 8\equiv 6\ (9)$ ma non è vero che $5\equiv 8\ (9)$

Quindi purtroppo operando con le classi di resto in generale non è vero che se $\bar{a} \cdot \bar{b} = \bar{a} \cdot \bar{c}$ allora $\bar{b} = \bar{c}$ come mostra l'esempio.

CIFRARI AFFINI

A questo punto possiamo già avere un'idea di come perfezionare la funzione di cifratura C_k.

Possiamo definire un'applicazione C_k che non solo contenga una traslazione ma anche una moltiplicazione. Allora

la nostra chiave sarà una coppia di numeri $k = (a,b)$

e la funzione cifrante sarà $C_k : Z_{21} \to Z_{21}$

$$p \to a \cdot p + b \mod 21,$$

Questo sistema prende il nome di **cifrario affine**.

Il problema è ora capire se questa funzione va bene, cioè dobbiamo vedere se ad ogni lettera dell'alfabeto in chiaro corrisponde una diversa lettera dell'alfabeto cifrante (perché questo ci assicura che, così, è possibile decifrare in modo univoco il testo) . Ovvero dobbiamo verificare che la funzione C_k è biunivoca.

Ricordiamo che una funzione tra due insiemi A (dominio) e B (condominio) è biunivoca se è iniettiva e suriettiva.

È iniettiva se ad elementi distinti di A corrispondono elementi distinti di B.

È suriettiva se ogni elemento di B è immagine di un elemento di A.

Nel nostro caso, quindi, dobbiamo verificare che :

- prese due lettere distinte dell'alfabeto in chiaro queste vengano criptate con lettere diverse (iniettività)
- ad ogni lettera dell'alfabeto cifrante resta associata una lettera dell'alfabeto in chiaro (suriettività).

Iniziamo con il vedere in dettaglio due casi:

scegliamo $k = (3,4)$ e $k = (5,4)$; il che equivale a dire che operiamo sempre la stessa traslazione, 4, ma come agente mescolante usiamo due numeri diversi, una volta 3 e un'altra 5.

Restano così definite le due applicazioni C_k:

la prima è $C_k : Z_{21} \to Z_{21}$

$$p \to 3 \cdot p + 4$$

la seconda è $C_k : Z_{21} \to Z_{21}$

$$p \to 5 \cdot p + 4$$

Visualizziamo in una tabella i risultati della cifratura:

Posiz. Lettera	Funz. 5p+4	Funz. 3p+4	Posiz. Lettera	Funz. 5p+4	Funz. 3p+4	Posiz. lettera	Funz. 5p+4	Funz. 3p+4
0 **a**	4	4	7 **h**	18	4	14 **q**	11	4
1 **b**	9	7	8 **i**	2	7	15 **r**	16	7
2 **c**	14	10	9 **l**	7	10	15 **s**	0	10
3 **d**	19	13	10 **m**	12	13	17 **t**	5	13
4 **e**	3	16	11 **n**	17	16	18 **u**	10	16
5 **f**	8	19	12 **o**	1	19	19 **v**	15	19
6 **g**	13	1	13 **p**	6	1	20 **z**	20	1

Si vede, quindi, che la funzione

$$C_k : Z_{21} \to Z_{21}$$
$$p \to 5 \cdot p + 4$$

è biunivoca, in quanto ad ogni lettera dell'alfabeto in chiaro resta associata una lettera diversa dell'alfabeto cifrato, ma questo non avviene con l'altra funzione

$$C_k : Z_{21} \to Z_{21}$$
$$p \to 3 \cdot p + 4$$

Allora possiamo concludere che in generale la funzione C_k non è biunivoca, ma per alcuni numeri la corrispondenza vale, per altri fallisce. È chiaro che non si può andare per tentativi nel decidere quali numeri possono andar bene e quali no, per questo sarà necessario studiare meglio le proprietà di Z_n e capire se esiste una regola generale per determinare, senza fare prove, se una funzione C_k è biunivoca, cioè se il sistema di cifratura che abbiamo formulato è realmente applicabile.

Terza lezione

Occorre ora definire bene le operazioni di congruenza e la struttura che definiscono.
Come sappiamo, se moltiplichiamo o dividiamo due numeri (che siano entrambi diversi da 1) o se li sommiamo o sottraiamo otteniamo due numeri differenti.
Se invece le nostre operazioni le facciamo modulo un numero a, allora otteniamo tutti i numeri compresi tra *0* e *a-1*, come abbiamo visto nelle lezioni precedenti.
Quando abbiamo analizzato la struttura del cifrario affine, abbiamo notato che sorgevano dei problemi.
Quando andavamo a traslare tutti i numeri di un certo valore *a* e poi li consideravamo di nuovo modulo 21, ottenevamo comunque tutti i numeri precedenti.
Calcolando invece i valori ottenuti da 5p+4 e 3p+4 in un caso ottenevamo tutti i numeri necessari, nell'altro no, alcuni numeri si ripetevano e altri erano assenti.
Da dove sorge tale problema?

Noi siamo abituati a fare tutti i nostri conti utilizzando l'insieme dei numeri *Reali*, che gode di una ben precisa proprietà:

$p \cdot q = 0$ se e solo se o $p = 0$ o $q = 0$

Sappiamo benissimo la formula precedente, la usiamo spesso, soprattutto per risolvere delle equazioni di secondo grado o superiori.

Tale formula è vera solo in parte nella nostra nuova struttura.
Se uno dei due tra *p* e *q* valgono 0 allora sicuramente il loro prodotto è 0 e resta tale anche modulo qualsiasi numero.
Quindi una parte dell'implicazione è perfettamente uguale al caso dei numeri reali.
La struttura di Z_n ha inoltre un'altra particolarità:
$p \cdot q = 0$ se e solo se o $p = 0$ o $q = 0$ o $p \cdot q = a \cdot n$

Infatti se $p \cdot q$ è un multiplo di *n* allora tale prodotto è 0 modulo *n*.
Abbiamo quindi 3 possibili casi per cui il prodotto tra *p* e *q* risulti nullo:

1. $p = 0$
2. $p = 0$
3. $p \cdot q =$ a un multiplo di *n*.

Logicamente se scegliamo come modulo *n* un numero primo torniamo al caso dei numeri reali, perdendo la terza opzione, perché comunque moltiplichiamo due numeri minori di n, non otterremo mai come risultato *n* perché *n* è un numero primo.

Osserviamo di nuovo le due tabelle ottenute dalla trasformazioni affini 5p+4 e 3p+4:

Posiz. Lettera	Funz. 5p+4	Funz. 3p+4	Posiz. Lettera	Funz. 5p+4	Funz. 3p+4	Posiz. lettera	Funz. 5p+4	Funz. 3p+4
0 a	4	4	7 h	18	4	14 q	11	4
1 b	9	7	8 i	2	7	15 r	16	7
2 c	14	10	9 l	7	10	15 s	0	10
3 d	19	13	10 m	12	13	17 t	5	13
4 e	3	16	11 n	17	16	18 u	10	16
5 f	8	19	12 o	1	19	19 v	15	19
6 g	13	1	13 p	6	1	20 z	20	1

Se consideriamo solo la traslazione 3p + 4 è evidente l'assenza di alcuni numeri mentre il numero 4 si ripete.
Non è un caso che sia proprio tale numero a ripetersi , e che si trovi sempre in relazione con i multipli di 7.
La prima volta che appare il quattro è in corrispondenza del numero 0, e siccome 4 è il valore della traslazione il risultato è esatto.
Il problema sorge dal fatto che stiamo facendo tutte le nostre operazioni modulo 21 = 3 x 7, e che abbiamo scelto come coefficiente per cui moltiplicare le lettere il numero 3.
In tale modo ogni volta che ci troviamo a moltiplicare per 3 il numero 7 e ogni suo multiplo, otterremo 21 o un suo multiplo, che quindi modulo 21 è 0.
Così otteniamo:

3 x 0 + 4 = 4 mod 21

3 x 7 + 4 = 21 + 4 = 4 mod 21

3 x 14 + 4 = 42 + 4 = 4 mod 21

Abbiamo perso così la corrispondenza biunivoca tra le lettere in ordine alfabetico e le lettere ottenute con la trasformazione affine.

Se consideriamo ora la trasformazione affine 5p + 4 vediamo dalla tabella che compaiono tutti e 21 i numeri.
Perché ciò accade se operiamo lo stesso modulo 21 che non è primo?
Tale risultato si ottiene perché 21 è divisibile solo per 3 e per 7, quindi se noi come coefficiente da moltiplicare nella trasformazione ne utilizziamo uno che non sia né 3 né 7 né un multiplo di uno dei due, non otterremo mai un multiplo di 21.
Per poter ottenere un multiplo di 21 moltiplicando le lettere per 5, dovremmo moltiplicare 5 per 21 o per un suo multiplo, e ciò significa moltiplicare 5 solo per 0, quindi solo la lettera *a* sarà messa in corrispondenza con il numero 4.

ESERCIZI
Completare la seguente tabella con le seguenti trasformazioni:

1. 7p +3
2. 5p + 21
3. 14p + 1
4. 21p + 1

Posiz. Lettera	Funz.	Funz.	Posiz. Lettera	Funz.	Funz.	Posiz. lettera	Funz	Funz.
0 a			7 h			14 q		
1 b			8 i			15 r		
2 c			9 l			15 s		
3 d			10 m			17 t		
4 e			11 n			18 u		
5 f			12 o			19 v		
6 g			13 p			20 z		

ALGORITMO EUCLIDEO e IDENTITA' DI BEZOUT

Ci proponiamo ora di fare una piccola digressione per introdurre uno strumento matematico che sarà fondamentale per lo sviluppo del corso.

Sappiamo tutti calcolare il MCD tra due numeri ragionevolmente piccoli (ad es. minori di 1000), di cui conosciamo (o possiamo facilmente calcolare) la scomposizione in fattori primi. Infatti ci è stato insegnato che, dati due numeri a,b e la loro scomposizione in primi, il $\mathrm{MCD}(a,b)$ si ottiene prendendo solo i fattori comuni con il minimo esponente.

L'algoritmo euclideo ci permette di poter calcolare il massimo comun divisore tra due numeri, anche se questi sono molto grandi, senza aver bisogno di fattorizzarli.

Riprendiamo alcune definizioni che dovrebbero essere note ed enunciamo l'unica proposizione su cui si fonda l'algoritmo euclideo.

Definizione Due numeri $a,b \in Z$ hanno sempre il MASSIMO COMUN DIVISORE cioè esiste sempre $d \in Z$, $d > 0$ tale che

1. d divide a e d divide b
2. se d' divide sia a che b allora d' divide d

E in tal caso si pone $\mathrm{MCD}(a,b) = d$

Definizione Due numeri interi a,b tali che $\mathrm{MCD}(a,b) = 1$ si dicono coprimi o relativamente primi.

Proposizione (divisione in Z) Siano $a,b \in Z$, $b \neq 0$. Allora esistono e sono univocamente determinati due interi q e r tali che

$a = b \cdot q + r$ con $0 \leq r < |b|$

Dimostrazione.

Consideriamo $b \geq 0$, e tutti i suoi multipli sia positivi che negativi, e possiamo quindi avere la successione:

…,-kb…-2b,-b,0,b,2b,….kb…. con $k>0$

Allora sicuramente esisteranno due multipli consecutivi di b tra cui è compreso a:

$qb \leq a < (q+1)b$

Poniamo $r = a - qb$ e così abbiamo trovato i due numeri q e b richiesti.

Inoltre loro sono unici perché q è il massimo numero tale che qb non supera a, e quindi anche r è unico per com'è definito, perché dipende solo da a,b e dall'unico numero q.

Com'è noto a tutti in quest'operazione a è detto *dividendo*, b *divisore*, q *quoziente* e r *resto*.

Da questa proposizione si nota subito che un numero a si dice divisibile per b se il resto della divisione è 0, e quindi, ovviamente, se esiste un numero q tale che $a = qb$.

Ogni numero a ha 4 divisori detti banali e sono: *a, -a, 1 e -1*. Se un numero ha per divisori solo quelli banali allora è un numero ***primo***.

L'algoritmo euclideo, che consente di calcolare il M.C.D. tra due qualsiasi numeri, si basa su divisioni successive: si inizia dalla divisione $a \div b$ e si ottengono un quoziente q_1 e un resto r_1, se $r_1 \neq 0$ si prosegue calcolando $b \div r_1$ e si ottengono q_2 e r_2, se $r_2 \neq 0$, si itera il ragionamento…

L'algoritmo termina quando troviamo resto nullo e il MCD che cerchiamo è l'ultimo resto diverso da zero. Questa procedura ha sicuramente termine perché al massimo dobbiamo compiere b passi prima di ottenere resto nullo. *(potremmo chiedere ai ragazzi perché e cercare di dimostrare insieme questa affermazione, oppure potremmo spiegarlo noi, oppure potremmo non dirlo affatto)*

Il procedimento da mettere in atto è illustrato schematicamente di seguito.

Dobbiamo calcolare $\mathrm{MCD}(a,b)$. Supponiamo $a \geq b > 0$ e operiamo le divisioni:

1	$a = b \cdot q_1 + r_1$	$0 < r_1 < b$
2	$b = r_1 \cdot q_2 + r_2$	$0 < r_2 < r_1$
3	$r_1 = r_2 \cdot q_3 + r_4$	$0 < r_4 < r_3$
	……	
i+2	$r_i = r_{i+1} \cdot q_{i+2} + r_{i+2}$	$0 < r_{i+2} < r_{i+1}$
	……	
n-1	$r_{n-3} = r_{n-2} \cdot q_{n-1} + r_{n-1}$	$0 < r_{n-1} < r_{n-2}$
n	$r_{n-2} = r_{n-1} \cdot q_n + r_n$	$0 < r_n < r_{n-1}$
n+1	$r_{n-1} = r_n \cdot q_{n+1} + 0$	

Allora $\mathrm{MCD}(a,b) = r_n$ (=ultimo resto non nullo).

Facciamo un esempio numerico per chiarire le idee.

ESEMPIO Calcolare $\mathrm{MCD}(3255,321)$

$3255 = 321 \cdot 10 + 312$
$321 = 312 \cdot 1 + 9$
$312 = 9 \cdot 34 + 6$
$9 = 6 \cdot 1 + 3$
$6 = 3 \cdot 2 + 0$

Allora $\mathrm{MCD}(3255,321) = 3$ (=ultimo resto non nullo)

Osservazioni.

1. Le divisioni che dobbiamo eseguire, che nel caso generale sono state segnate con n+1, sono al massimo pari al numero b, perché ogni resto r_i delle divisioni è minore del resto precedente, quindi abbiamo una catena $b>r_1>r_2>r_3>\ldots$ di numeri che sono interi positivi e decrescenti, quindi hanno un minimo che è 0 e sono al massimo b.
2. Il divisore comune di *a* e *b* che stiamo cercando, è lo stesso di *b* e r_1, perché se un numero divide *a* e *b* allora ne divide anche la differenza, cioè proprio r_1. Tale ragionamento vale per tutti i resti delle divisioni successive fino al resto r_n che è l'ultimo diverso da 0.
3. Gli ultimi due resti r_{n-1} e r_n sono multipli, quindi i divisori che hanno in comune sono i divisori di r_n che è il numero più piccolo.
4. L'ultimo resto diverso da 0, r_n, oltre ad essere un divisore comune sia ad *a* che a *b* è anche il più grande. Infatti, poniamo $r_n = d$, se esistesse un altro divisore *d'* dei numeri *a* e *b*, allora *d'* dividerebbe anche *d*. Quindi *d*, ovvero r_n, è effettivamente il massimo comun divisore.

Un ultimo caso da osservare è se *b* è un numero negativo.
In tale caso come prima si avrà:
$a = -b \cdot q' + r$ con $0 \leq r < |b| = -b$

Basta poi porre $q' = -q$ per trovare poi i due numeri *q* e *r* richiesti.

L'algoritmo di Euclide ci permette, una volta individuato $d = \text{MCD}(a,b)$, di trovare due numeri interi s,t tali che

$d = s \cdot a + t \cdot b$

questa relazione si chiama **IDENTITA' DI BEZOUT**.

Per dimostrarla basta far vedere che tutti i resti delle divisioni successive si possono scrivere come combinazioni di a e b. Infatti osserviamo che:

$r_1 = a - b \cdot q_1$

$r_2 = b - r_1 \cdot q_2$

....

$r_{i+2} = r_i - r_{i+1} \cdot q_{i+2}$

....

$r_n = r_{n-2} - r_{n-1} \cdot q_n$

da cui

$$r_2 = b - r_1 \cdot q_2 = b - (a - b \cdot q_1)q_2 = (1 + q_1 q_2)b - q_2 a$$

cioè r_1 e r_2 si scrivono come combinazione di *a* e *b* ovvero come un multiplo intero di a più un multiplo intero di b. Ma allora anche r_3, che è una combinazione a coefficienti interi di r_1 e r_2, sarà una combinazione a coefficienti interi di a e b. In definitiva, $d = r_n$ è una combinazione a coefficienti interi di r_{n-1} e r_{n-2} e quindi di a e b.

Notiamo che l'espressione del MCD(a,b) fornita dall'identità di Bezout non è affatto unica. Ad esempio: $1 = 3 \cdot 7 + (-4) \cdot 5 = (-2) \cdot 7 + 3 \cdot 5$

Utilizzando l'esempio precedente, vediamo come procedere per trovare un'identità di Bezout.
Dobbiamo individuare $s,t \in Z$ tali che $3 = s \cdot 3522 + t \cdot 321$.
Riscriviamo i passaggi dell'algoritmo euclideo nel modo seguente:

$r_1 = 312 = 3255 - 321 \cdot 10$

$r_2 = 9 = 321 - 312 \cdot 1$

$r_3 = 6 = 312 - 9 \cdot 34$

$r_4 = 3 = 9 - 6 \cdot 1$

Partiamo dall'ultima relazione scritta, sostituiamo il numero esplicitato nell'equazione precedente, raccogliamo i fattori comuni e continuiamo a sostituire fino ad ottenere un'espressione in nei numeri a,b. Ovvero:

$3 = 9 - 6 \cdot 1 =$

$= 9 - (312 - 9 \cdot 34) \cdot 1 =$

$= 9 - 312 \cdot 1 + 9 \cdot 34 \cdot 1 = 35 \cdot 9 - 312 =$

$= 35 \cdot (321 - 312 \cdot 1) - 312 =$

$= 321 \cdot 35 - 312 \cdot 1 \cdot 35 - 312 = 35 \cdot 321 - 36 \cdot 312 =$

$= 35 \cdot 321 - 36 \cdot (3522 - 321 \cdot 10) =$

$= 35 \cdot 321 - 36 \cdot 3522 + 36 \cdot 321 \cdot 10 = 395 \cdot 321 - 36 \cdot 3522$

Abbiamo così ottenuto che $3 = (-36) \cdot 3522 + (395) \cdot 321$, cioè $s = -36$ e $t = 395$.

ESEMPIO Calcolare MCD(44880,5292) e la successiva identità di Bezout.

$44880 = 5292 \cdot 8 + 2544$

$5292 = 2544 \cdot 2 + 204$

$2544 = 204 \cdot 12 + 96$

$204 = 96 \cdot 2 + 12$

$96 = 12 \cdot 8 + 0$

Quindi MCD(44880,5292)=12

Ora ricaviamo identità di Bezout, notando che non è necessario riscrivere i passaggi dell'algoritmo euclideo:

$12 = 204 - 96 \cdot 2 = 204 - (2544 - 204 \cdot 12) \cdot 2 = 204 - 2544 \cdot 2 + 204 \cdot 24 =$

$= 25 \cdot 204 - 2 \cdot 2544 = 25 \cdot (5292 - 2544 \cdot 2) - 2544 \cdot 2 =$

$= 5292 \cdot 25 - 2544 \cdot 50 - 2544 \cdot 2 =$

$= 5292 \cdot 25 - 2544 \cdot 52 = 5292 \cdot 25 - (44880 - 5292 \cdot 8) \cdot 52 =$

$= 5292 \cdot 25 - 44880 \cdot 52 + 5292 \cdot 416 =$

$= 5292 \cdot 441 - 52 \cdot 44880$

Quindi abbiamo ottenuto $12 = (-52) \cdot 44880 + (441) \cdot 5292$, ovvero $s = -52$ e $t = 441$.

ESERCIZI. Calcolare il MCD e la relativa identità di Bezout tra le seguenti coppie di numeri

1. 126 e 35 (7=2*126-7*35)
2. 44275 e 605 (55=(-1)*44275+74*605)
3. 2470 e 351 (13=2470-7*351)

Quarta lezione

Nella lezione precedente abbiamo imparato che dato un massimo comun divisore tra due numeri possiamo calcolarne la relativa identità di Bezout.
Tale procedimento però ci consente anche di poter risolvere un equazioni a due incognite:

$ax + ny = b$,

Tale equazione si può considerare derivare dalla congruenza:

$ax = b \bmod n$.

In tale modo otteniamo :

$ax - ny = b \bmod n$.

Teorema La congruenza $ax = b \bmod n$ ammette una soluzione se e solo se il M.C.D.(a, n) divide b.

Dimostrazione.
Se $ax = b \bmod n$ allora $ax - b = 0 \bmod n$. Questo significa che n divide $ax - b$.
Ovvero che o n o un suo multiplo è pari a $ax - b$:

$ax - b = ny$ ovvero $ax - ny = b$ e ponendo $-y = t$ otteniamo $ax + ty = b$.

Dunque ora devo dimostrare che $ax + ty = b$ se e solo se M.C.D.(a , n) divide b.
Siccome dobbiamo dimostrare un se e solo se dividiamo la dimostrazione in due parti.

Supponiamo che la soluzione esiste e verifichiamo la condizione dull' M.C.D.
Siano x e t una soluzione, e d = M.C.D.(a,n), allora d divide ax e ny, quindi divide anche al loro somma cioè $ax + by = b$, cioè d divide b.

Supponiamo che d = M.C.D.(a,n) e che d divida b, ossia che esiste un valore h tale che $b = dh$, e verifichiamo che l'equazione ammette una soluzione. Posso calcolare d con l'algoritmo Euclideo e poi scrivere la relativa identità di Bezout.
Troverò così dei valori tali che $\alpha a + \beta n = d$, moltiplicando ambo i membri per h otteniamo:

$h\alpha a + h\beta n = dh = b$.

Basta ora porre $h\alpha = x$ e $h\beta = t$ e abbiamo ottenuto le soluzioni cercate.

Si può inoltre dimostrare che il "numero " delle soluzioni di tale equivalenza è esattamente pari al M.C.D.(a, n).
Data una soluzione x, tutte le altre si ottengono con la formula $x + mh$ dove $m = n/d$ e
$h = 1, 2, \ldots, d-1$.
Siccome h può variare tra un insieme di d numeri, le soluzioni sono solo d.
Inoltre se h supera il valore di d si ottengono soluzioni congrue tra loro.

Dunque, ad esempio, data l' equazione:
$5x + 6y = 1$
Il primo passo è verificare se M.C.D.(5 , 6) = 1 divide 1.
Calcolare con l'algoritmo Euclideo il M.C.D. (5, 6).
Ottenere la relativa identità di Bezout.
Trovare i due numeri x, y richiesti.
Nel caso in cui il M.C.D. ottenuto è maggiore di 1, calcolare tutte le possibili soluzioni.

ESERCIZI Trovare le soluzioni delle seguenti equazioni:

$92x + 28y = 180$

$482+x\ 20y = 35$

$1859x + 2057y = 143$

$180x + 27y = 3$

CIFRARI POLIALFABETICI

Utilizzare un solo alfabeto per cifrare rende il compito del crittoanalista non troppo difficile.
Il primo ad intuire tale affermazione fu Leon Battista Alberti, che nella seconda metà del 1400 propose di utilizzare due alfabeti per cifrare.
Tale idea fu ampliata e completata da Vigenère, che invece che limitarsi nell'utilizzo di soli due alfabeti, decise di usarne 26, uno per ogni lettera dell'alfabeto.
In questo modo ogni singola lettera della frase da cifrare viene crittata con un differente alfabeto, secondo una regola che viene decisa dalla tavola di Vigenère e dalla parola chiave.
Anche in questo caso la parola chiave da utilizzare deve essere concordata precedentemente tra chi spedisce il messaggio e chi lo riceve, e deve assolutamente rimanere segreta tra loro due.
Supponiamo che le due persone che desiderano scambiarsi i messaggi decidano di usare come parola chiave "algebra".
Il testo che deve essere nascosto è la frase: " assediare la città".
Logicamente nel nostro alfabeto tutte le lettere accentate non contano, quindi è come se dovessimo cifrare "assediare la citta".
La prima cosa da fare è scrivere tante volte la parola chiave quanto è lungo il testo da cifrare:

a l g e b r a a l g e b r a a l
a s s e d i a r e l a c i t t a.

In questo modo ad ogni lettera del testo da cifrare viene associata una lettere della parola chiave.
La tavola di Vigenère crea una corrispondenza tra le lettere in chiaro e le lettere cifrate.

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
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	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	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

Procedimento:

- nella prima colonna va cercata la lettera del testo da cifrare
  (ad esempio al posto x di questa colonna)
- si vede quale è la lettera della parola chiave associata
- si trova tale lettera nella prima riga
  (ad esempio al posto y di tale riga)
- si trova la lettera che si trova alla posizione x delle colonne e y delle righe, ed essa cifrerà la prima lettera del testo

Il procedimento va ripetuto per tutte le lettere del testo.

Nel nostro caso la A di assediare è associata alla lettera A di algebra, e quindi viene cifrata in A.
La lettera S di assediare è associata alla lettera L di algebra, e quindi bisogna prendere sulla prima colonna il posto dove compare s, sulla prima riga il posto dove compare l e trovare la lettera D.
Ripetendo il procedimento per tutte le lettere otteniamo:

a l g e b r a a l g e b r a a l
a s s e d i a r e l a c i t t a
A D Y I E Z A R P R E D Z T T L

E' evidente che, a differenza dei cifrari monoalfabetici, a lettere uguale non viene associata più una lettera uguale.
In tale modo l'analisi delle frequenze non è più applicabile.
Sembrerebbe che sia fondamentale la conoscenza della parola chiave.
Non serve in realtà conoscerla in maniera assoluta, per far iniziare a crollare tale sistema è sufficiente conoscere da quanti caratteri essa è costituita.

Il punto debole di tale sistema è che è equivalente a n cifrari di Cesare differenti, dove n è il numero di lettere che compongono la parola chiave.
Il punto debole di un cifrario monoalfabetici è l'analisi delle frequenze.
Unendo queste due osservazioni si giunge a una perfetta crittoanalisi di tale cifratura.
Il primo a decifrare tale codice fu un crittoanalista tedesco F.W. Kasiski, verso la fine del 1800, per tale motivo questo metodo non viene più preso in considerazione, anche se garantisce un certo margine di sicurezza.

La nostra parola chiave è "algebra", composta da sole sette lettere, questo significa che se dobbiamo crittare due lettere uguali che si trovano in posizioni x e y e tali che x non è congruo y modulo 7, allora le precedenti due lettere saranno cifrate in due differenti modi.
Il problema sorge ogni qual volta le due lettere uguali da cifrare si trovano poste a distanza di sette lettere, o di un multiplo di sette.
Nel nostro caso era evidente come la prima lettera e l'ultima lettera A della parola assediare si trovano a sette lettere di distanza, infatti entrambi le A vengono cifrate con la stessa lettera, che solo casualmente è ancora una A.
Un buon rimedio a tale problema consiste nello scegliere che invece una singola parola chiave, una più lunga "frase chiave",composta da molte più lettere che consente di regalare al crittoanalista un lavoro un po' più duro.
Possiamo sostituire "algebra" con "usiamo algebra come parola chiave"

Un ulteriore problema deriva anche da come si sceglie la parola chiave, nel nostro esempio la prima e l'ultima lettera di tale parola sono le stesse, e nella ripetizione di tale parola avviene la coincidenza che tali lettere vengono associate alle doppie T della parola città.
Tale avvenimento fa in modo che di nuovo due lettere uguali vengono crittate con altre due lettere uguali.
Sarebbe quindi utile evitare di ottenere simili coincidenze, un ulteriore metodo consiste nell'eliminare tutte le lettere uguali.
La nostra precedente frase chiave "usiamo algebra come parola chiave" diventa quindi:
"usiamo lgebr c p hiav".

Tutti questi rimedi non eliminano il problema principale, la lunghezza della chiave.
Se un chiave è costituita da *n* caratteri, allora ogni *n* caratteri le singole lettere vengono cifrate con lo stesso cifrario di Cesare.
Quindi di nuovo a lettera uguale corrisponde cifratura uguale.
La soluzione di tale problema si ottiene calcolando le singole frequenze delle lettere cifrate considerando però la lunghezza della parola chiave.
Si ritorna così al caso di un singolo cifrario di Cesare, si procede per tentativi considerando tutti e n i cifrari (dove n è al massimo 26) e in tale modo si ottiene la cifratura del testo nascosto.

ESERCIZI Utilizzando il metodo di Vigenère crittare un testo scelto, utilizzando:

- una parola chiave
- una frase chiave
- una frase chiave senza lettere ripetute

Quinta lezione

SISTEMI A CHIAVE PUBBLICA

Finora abbiamo descritto dei metodi di cifratura che fanno uso di una chiave privata, cioè di una informazione indispensabile sia per criptare il messaggio che per decifrarlo e quindi tale chiave deve essere necessariamente nota sia al mittente che al destinatario.
E' ovvio però che quest'ultimo deve essere l'unico a conoscere questa informazione, altrimenti chiunque altro sarebbe in grado di cifrare i messaggi con lo stesso metodo, ma soprattutto di scoprire ciò che invece si vuole tenere segreto!

Immaginiamo ora che il destinatario voglia comunicare privatamente con più di una persona, anzi che voglia addirittura che chiunque sia in grado di inviargli messaggi cifrati, mantenendo però la segretezza di ciascuno. Con i sistemi a chiave privata finora utilizzati ciò che auspichiamo è in pratica irrealizzabile, perché se il destinatario stabilisce una sua chiave e la diffonde chiunque è in grado di inviargli messaggi cifrati, ma è ovvio che ciascun mittente può facilmente leggere ciò che scrive ogni altro mittente.
Ed è facilmente intuibile che accordare una chiave diversa per ogni mittente può, nella maggior parte dei casi che comunemente si verificano, essere scomodo, rischioso se non addirittura impraticabile.

E' necessario quindi un sistema diverso, un metodo che si basi su due informazioni: una nota a tutti (e che viene comunemente chiamata *chiave pubblica*) necessaria per cifrare e un'altra (la *chiave privata*) che conosce solo il destinatario e permette a lui soltanto di leggere i messaggi.
Questo sistema fu ideato per la prima volta da Diffie e Hellman, ma lo realizzarono per primi Rivest, Shamir e Adleman del MIT (Massachusetts Institute of Tecnology), e in loro onore prende il nome di *sistema RSA*.

Prima di addentrarci nel procedimento matematico e pratico vero e proprio, proviamo a spiegare il concetto che si nasconde dietro questo sistema attraverso un esempio semplice.
Immaginiamo che A voglia spedire a B un oggetto in una scatola e voglia fare in modo che nessuno possa guardarne il contenuto. Può chiudere la scatola con un lucchetto, ma B deve possedere la chiave per aprirlo e quindi è necessario che A spedisca a B prima la chiave e solo successivamente la scatola chiusa. Ma questo procedimento richiede tempi lunghi e inoltre c'è il rischio che la chiave possa andare perduta o peggio ancora essere intercettata da qualcun altro (e così avremmo operato col vecchio cifrario a chiave privata).
Il sistema RSA funziona così invece: B mette a disposizione di chiunque voglia spedirgli qualcosa un lucchetto di cui già possiede la chiave (e soltanto lui la possiede!), A si procura la "chiave pubblica" di B cioè chiude la scatola con questo lucchetto e la spedisce...e B (e nessun altro) sarà in grado di aprirla!

L'idea concettuale che si cela dietro questo potente sistema crittografico è geniale: ci è permesso divulgare sia la chiave di cifratura che il metodo, ma senza per questo rivelare contestualmente il modo di decifrare. Il "lucchetto" pubblico è in realtà una funzione matematica abbastanza semplice che tutti sono in grado di utilizzare, mentre la funzione inversa consente di tornare indietro agevolmente solo a chi è in possesso della "chiave". Il tutto è quindi basato su una funzione cifrante, la cui inversa è complessa solo apparentemente e diventa improvvisamente molto semplice non appena la si guarda attraverso l'informazione aggiuntiva (data dalla chiave).

Per questo motivo i sistemi a chiave privata sono detti anche *simmetrici,* mentre quelli a chiave privata *asimmetrici.*
Scopriremo più avanti di quale funzione stiamo parlando e capiremo nelle prossime lezioni quale dato riveste il ruolo della chiave e perché questa informazione sia (almeno ad oggi) indispensabile.

Tuttavia, per quanto la funzione sia semplice concettualmente, la cifratura e la decifratura richiedono conti abbastanza complessi e non molto agevoli da trattare se non si usa il metodo migliore: ci imbatteremo, infatti, nel calcolo di congruenze in cui compaiono potenze con basi ed esponenti elevati.
E' vero che, solitamente, questi conti vengono svolti dai calcolatori (anche perché i numeri coinvolti sono costituiti da tantissime cifre e quindi non sono assolutamente trattabili a mano), ma cerchiamo comunque di capire (usando ovviamente cifre più piccole) come il computer lavora per calcolare queste potenze.

SISTEMA BINARIO

Introduciamo un concetto forse già parzialmente noto, almeno a chi ha un minimo di conoscenza del calcolatore: la numerazione in base 2.

Iniziamo col considerare il numero 3013: questa scrittura in realtà sappiamo che implicitamente vuol dire $3 \cdot 10^3 + 0 \cdot 10^2 + 1 \cdot 10 + 3 \cdot 10^0$ poiché noi utilizziamo per esprimere i numeri dieci cifre (da 0 a 9) in una specifica posizione che da alla cifra usata un significato. Ad esempio in questo esempio abbiamo usato le cifre 0,1 e 3 e quest'ultima ripetuta due volte, ma è ovvio che il primo 3 rappresenta un numero diverso dal secondo in quanto occupa una posizione diversa o, detto in altri termini, il primo è coefficiente di una potenza di 10 diversa dal secondo.

Se guardiamo più attentamente, però, ci accorgiamo che dividendo 3013 per 10 e poi procedendo ancora per divisioni successive (dividendo il quoziente ottenuto sempre per 10) fin quando non otteniamo un quoziente nullo si ha:

$$\begin{aligned} 3013 &= 10 \cdot 301 + 3 \\ 301 &= 10 \cdot 30 + 1 \\ 30 &= 10 \cdot 3 + 0 \\ 3 &= 10 \cdot 0 + 3 \end{aligned}$$

e i resti ottenuti sono proprio le cifre (lette dal basso verso l'alto) che compongono il numero 3013.

Ci chiediamo: è possibile utilizzare lo stesso procedimento per basi diverse da 10? La risposta è affermativa: in realtà la scelta della base 10 è puramente convenzionale, ma anche storicamente si incontrano numerazioni diverse. I Babilonesi, ad esempio, usavano la base 60, i Maya la base 20 e i calcolatori usano la base 2, ossia utilizzano solo due cifre per rappresentare i numeri: 0 e 1.

Il procedimento utilizzato per qualsiasi base *b* è lo stesso: il ruolo svolto da 10 è svolto da *b* e quindi le cifre utilizzate nella rappresentazione sono quelle che vanno da 0 a *b-1,* le potenze coinvolte sono quelle di *b* e i coefficienti si calcolano per divisioni successive per *b* e sono date dai resti di queste divisioni.

Vediamo un esempio pratico di come passare dalla base 10 alla base 2 (che è quella che a noi interessa).
Consideriamo il numero 637 (in base 10) e svolgiamo le divisioni successive per 2:

$$\begin{aligned} 637 &= 2\cdot 318 + 1 \\ 318 &= 2\cdot 159 + 0 \\ 159 &= 2\cdot 79 + 1 \\ 79 &= 2\cdot 39 + 1 \\ 39 &= 2\cdot 19 + 1 \\ 19 &= 2\cdot 9 + 1 \\ 9 &= 2\cdot 4 + 1 \\ 4 &= 2\cdot 2 + 0 \\ \\ 1 &= 2\cdot 0 + 1 \end{aligned}$$

Come prima, ci siamo fermati una volta trovato un quoziente nullo.
I resti, letti dal basso verso l'alto, ci forniscono la rappresentazione del numero 637 in base 2:

$$(637)_{10} = (1001111101)_2$$

Viceversa, se vogliamo passare dalla base 2 alla base 10, non dobbiamo fare altro che scrivere per esteso il numero moltiplicando ciascuna cifra per la relativa potenza di 2 (come avevamo fatto all'inizio per 3013 con la base 10):

$$(1001111101)_2 = 1\cdot 2^9 + 0\cdot 2^8 + 0\cdot 2^7 + 1\cdot 2^6 + 1\cdot 2^5 + 1\cdot 2^4 + 1\cdot 2^3 + 1\cdot 2^2 + 0\cdot 2^1 + 1\cdot 2^0 =$$

$$= 512 + 64 + 32 + 16 + 8 + 4 + 1 = 637\,.$$

Torniamo, infine, al punto da cui siamo partiti: come questo cambiamento di base ci consente di calcolare il valore modulo *n* di un numero elevato ad un esponente grande.
Supponiamo di voler calcolare $35^{637} \mod 382$: di certo non possiamo elevare 35 alle 637 perché l'operazione richiederebbe (oltre all'utilizzo di un calcolatore) un tempo impossibile!
Sfruttiamo la scrittura di 637 in base 2 (che abbiamo appena calcolato):

637=1001111101=512+64+32+16+8+4+1
e quindi $35^{637} \mod 382 = 35^{512}\cdot 35^{64}\cdot 35^{32}\cdot 35^{16}\cdot 35^{8}\cdot 35^{4}\cdot 35^{1} \mod 382$
e apparentemente il problema non sembra semplificato perché dobbiamo ancora calcolare delle potenze con esponenti abbastanza elevati come 512, 64, 32, 16, ecc.

Ma $35^2 \equiv 79 \mod 382$ e possiamo proseguire con le potenze di 2 e calcolare $35^4 \mod 382$: basta osservare che $35^4 \equiv 35^2\cdot 35^2 \mod 382$ e quindi, sfruttando quello che abbiamo appena calcolato, $35^4 \equiv 79^2 \equiv 129 \mod 832$.
Procedendo con lo stesso sistema si ottegono:
$35^8 \equiv 129^2 \equiv 215 \mod 832$
$35^{16} \equiv 215^2 \equiv 3 \mod 832$
$35^{32} \equiv 3^2 \equiv 9 \mod 832$

$35^{64} \equiv 9^2 \equiv 81 \mod 832$
$35^{128} \equiv 81^2 \equiv 67 \mod 832$
$35^{256} \equiv 67^2 \equiv 287 \mod 832$
$35^{512} \equiv 287^2 \equiv 239 \mod 832$

Quindi $35^{637} \mod 832 = 239 \cdot 81 \cdot 9 \cdot 3 \cdot 215 \cdot 129 \cdot 35 \mod 832$ e a questo punto si può risolvere la congruenza moltiplicando opportunamente due fattori per volta e riducendoli di volta in volta modulo 832. Il risultato finale è $35^{367} \equiv 121 \mod 832$.

Questo metodo è efficace per semplificare il problema, perché non abbiamo mai calcolato in pratica potenze con esponenti grandi, ma abbiamo di volta in volta elevato solo al quadrato.
Possiamo, quindi, giungere al risultato finale senza calcolare realmente la potenza.

ESERCIZI Calcolare col metodo appena visto le seguenti potenze modulo n=92:

- 7^{237}
- 24^{53}

Scopriamo adesso, come promesso, quale sarà la funzione che impareremo ad usare nell'RSA:

Definizione *Funzione di Eulero*
Per ogni $n \geq 1$, si indica con $\varphi(n)$ il numero di interi positivi minori di *n* e relativamente primi con *n.*
Tale funzione prende il nome di funzione di Eulero di *n*.

ESEMPIO
$\varphi(16) = 8$, infatti i numeri minori di 16 e relativamente primi con 16 sono: 1, 3, 5, 7, 9, 11, 13, 15.

Definizione Una funzione *f* si dice *moltiplicativa* se

$$f(rs) = f(r)f(s) \qquad \text{per ogni r,s primi tra loro}$$

Una delle proprietà più importanti della funzione di Eulero è proprio che è moltiplicativa:

$$\text{se } n=pq \qquad \text{allora} \quad \varphi(n) = \varphi(p)\varphi(q)$$

Proviamo a dimostrare quest'ultima affermazione:
$\varphi(n) = \varphi(pq)$ è dato per definizione da tutti gli elementi compresi tra 0 e pq-1 relativamente primi con *pq.* Quindi per contare quanti sono questi elementi dobbiamo sottrarre a *pq* (cioè il numero totale di elementi considerati) il numero di quelli che non sono coprimi con *pq*, ovvero i numeri non divisibili per *p* e/o per *q.*
I multipli di *p* compresi tra 0 e *pq-1* sono: 0=0*p, p=1p, 2p, 3p,…, (q-1)p* e quindi in totale sono esattamente *q* elementi.
Analogamente per *q:* 0=0*q, q=1q, 2q, 3q,…,(p-1)q*, quindi *p* numeri.

Osserviamo infine che lo 0 è multiplo di entrambi e quindi va contato una sola volta: togliendo, quindi, a pq il numero di elementi multipli di p e di q ne va aggiunto uno in modo da togliere lo 0 soltanto una volta.

Quindi, in conclusione,
$\varphi(n) = \varphi(pq) = pq - q - p + 1 = q(p-1) - (p-1) = (p-1)(q-1) = \varphi(p)\varphi(q)$

Ma come calcoliamo a questo punto $\varphi(p)$ e $\varphi(q)$ *? Discussione*

Se guardiamo la definizione, basta osservare che i numeri minori di p, ad esempio, e relativamente primi con p sono tutti quelli compresi tra 1 e $p-1$ e quindi

$$\varphi(p) = p - 1$$

Tali proprietà ci permettono di calcolare facilmente la funzione φ di un qualsiasi n, quando conosciamo la sua scomposizione in fattori primi.

ESEMPI
se $p=11$, $\varphi(11) = 10$
se $q=17$, $\varphi(17) = 16$
quindi se $n=11\times17=187$, $\varphi(187) = \varphi(11)\varphi(17) = 10 \cdot 16 = 160$.

In generale si può dimostrare (ma noi lo enunceremo soltanto) che la moltiplicatività di φ vale anche quando n è prodotto di più di due primi, anche se vengono presi più volte:

Proposizione: Se $n = p_1^{h_1} p_2^{h_2} \cdots p_s^{h_s}$, con p_i primi distinti per $i = 1,\ldots s$, allora

$$\varphi(n) = \varphi(p_1^{h_1})\varphi(p_2^{h_2}) \cdots \varphi(p_s^{h_s})$$

e se p è un numero primo, allora:

$$\varphi(p^h) = p^h - p^{h-1}$$

Osservazione.
Riprendiamo per un attimo il concetto di elementi invertibili di Z_n: questi erano tutti e soli gli elementi relativamente primi con n.
Non è quindi difficile convincersi che qualunque sia n il numero di tali elementi invertibili è esattamente $\varphi(n)$.
Ad esempio, se $n=8$, $\varphi(8) = 4$ e quindi il numero degli elementi invertibili di Z_8 è 4: $\bar{1},\bar{3},\bar{5},\bar{7}$.

Dimostriamo adesso due risultati importanti e piuttosto famosi sulle congruenze, a cui per necessità di dimostrazione premettiamo un lemma:

Lemma: Sia p è un numero primo ed $a \neq 0$ non divisivile per p, allora $a^{p-1} \equiv 1 \mod (p)$.

Dimostrazione.
Consideriamo l'insieme formato dagli elementi

$$\{a \bmod p,\ 2a \bmod p, \ldots, (p-1)a \bmod p\}.$$

Questi elementi sono tutti distinti: prendiamone due qualsiasi, ra ed sa e supponiamo $ra=sa$.
Sappiamo che, poiché a è relativamente primo con p, è invertibile in Z_p e quindi avrà un inverso che chiamiamo b.
Moltiplicando entrambi gli elementi per b si ha $r=s$.
Inoltre nessuno degli elementi dell'insieme può essere 0: supponiamo infatti che $ka=0$ per qualche k. Allora per definizione di congruenza, p deve dividere ka, ma ciò è impossibile perché p non divide a per ipotesi e non divide k perché sempre per ipotesi $k<p$.
Abbiamo quindi un insieme di $p-1$ elementi distinti, minori di p (perché sono tutti ridotti modulo p) e diversi da zero: tale insieme deve quindi coincidere con $\{1,2,\ldots,p-1\}$.
Quindi anche il prodotto degli elementi del primo insieme deve essere uguale al prodotto di quelli del secondo:

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

da cui segue (moltiplicando entrambi i membri per l'inverso di $(p-1)!$) la tesi.

Piccolo teorema di Fermat: Sia a un intero e p un numero primo. Allora:

$$a^p \equiv a \bmod p$$

Dimostrazione.
Se $a=0$ allora il risultato è ovvio. Se $a \neq 0$ allora possiamo applicare il lemma precedente e moltiplicare entrambi i membri per a [*perché si può fare? Lo facciamo dire a loro]*

Teorema di Eulero: Se a è relativamente primo con n, allora

$$a^{\varphi(n)} \equiv 1 \bmod n$$

Dimostrazione
Sia $n=pq$, dal piccolo teorema di Fermat sappiamo che

$$a^{p-1} \equiv 1 \mod (p) \quad e \quad a^{q-1} \equiv 1 \mod (q)$$

ma poiché p e q sono primi, $p-1$ e $q-1$ sono le funzioni di Eulero di p e q, quindi

$$a^{\varphi(p)} \equiv 1 \bmod p \qquad e \qquad a^{\varphi(q)} \equiv 1 \bmod q$$

Ricordando che $\varphi(n) = \varphi(p)\varphi(q)$, possiamo elevare entrambi i membri di queste ultime congruenze rispettivamente alla $\varphi(n)/\varphi(p)$ e alla $\varphi(n)/\varphi(q)$:

$$a^{\varphi(n)} \equiv 1 \bmod (p) \quad \text{e} \quad a^{\varphi(n)} \equiv 1 \bmod (q)$$

Quindi $a^{\varphi(n)} \equiv 1 \bmod (pq)$, cioè modulo *n*.

Finalmente nella prossima lezione vedremo come applicare questi risultati teorici per ricostruire il sistema RSA e impareremo a cifrare e decifrare.

Sesta lezione

RSA

Cifratura

Riprendiamo ora le "scatole" e i "lucchetti" della lezione precedente e vediamo come utilizzarli nella pratica: passiamo, cioè, a scoprire la cifratura.
Prima di tutto per usufruire del sistema è necessario procurarsi una chiave pubblica da iscrivere in un elenco di dominio pubblico, al quale potrà attingere chiunque voglia scriverci.
La chiave è costituita da due numeri, che indicheremo con *n* ed *e*, che possiamo scegliere come vogliamo purché *n* sia prodotto di due numeri primi molto grandi *p* e *q* ed *e* sia un qualsiasi numero relativamente primo con *p-1* e *q-1* (le motivazioni di questa richiesta ci saranno chiare in seguito).

Quindi il destinatario, che chiameremo per comodità ancora B, deve pubblicare la sua chiave *(n,e)* generando due numeri primi *p* e *q* molto grandi, moltiplicandoli tra loro per ottenere *n* e scegliendo un qualsiasi intero *e* relativamente primo con *p-1* e *q-1* ed il gioco è fatto.
L'unica accortezza che deve avere è di tenere nascosti i primi *p* e *q* che, come vedremo, saranno la sua chiave privata per decifrare.
Verso la fine del laboratorio ci soffermeremo un po' sul generare i numeri primi, su come sia possibile farlo e sulla relativa semplicità che i calcolatori ci garantiscono.

Spieghiamo invece adesso, col supporto di un esempio, come deve procedere il mittente A se vuole inviare un messaggio a B.
A vuole cifrare a B la parola **ALGEBRA;** prima di tutto A deve cercare nell'elenco pubblico la chiave di B: supponiamo che sia *(n=1003, e=3).*
Adesso deve trasformare ogni lettera in un intero, ricordando che (usando l'alfabeto inglese) la lettera *a* corrisponde a 00 e la lettera *z* a 25.
La sequenza che ottiene è 00 11 06 04 01 17 00.
Successivamente A suddivide il messaggio in blocchi in modo tale da ottenere interi minori di *n* e relativamente primi con *n* (anche il motivo di queste ipotesi ci sarà chiaro più avanti): nel nostro esempio si può suddividere la parola in blocchi di due lettere (o digrafi) aggiungendo tante X quante sono necessarie per completare i blocchi:

AL	GE	BR	AX
0011	0604	0117	0023

Infatti i numeri $P_1 = 11$, $P_2 = 604$, $P_3 = 117$, $P_4 = 23$ sono minori di *n* e coprimi con *n* (si verifica con l'algoritmo euclideo).
Si può verificare anche che dividendo in blocchi di tre lettere, si otterrebbero interi maggiori di *n*.

A questo punto A è pronto per la cifratura vera e propria: l'operazione principale consiste nell'elevare i P_i alla *e* e ridurli modulo *n* => $C_i = P_i^e \mod n$.

Proseguendo col nostro esempio, dobbiamo calcolare quindi $11^3 \mod 1003$, $604^3 \mod 1003$, $117^3 \mod 1003$ e $23^3 \mod 1003$ =>

$$C_1 = 328,\ C_2 = 797,\ C_3 = 825,\ C_4 = 131$$

Ora A è pronto ad inviare il messaggio a B: gli basterà mandargli i C_i che ha appena calcolato e poi spetterà a B decifrare!

Decifratura

Procedendo col nostro esempio, il destinatario ha ricevuto delle cifre, che abbiamo indicato con C_i, che sono $C_1 = 328,\ C_2 = 797,\ C_3 = 825,\ C_4 = 131,$ tali che $C_i = P_i^e \bmod(n)$ e deve risalire al messaggio originario, cioè ai P_i.
Abbiamo bisogno di determinare la nostra "chiave segreta", che indichiamo con d, sfruttando la nostra informazione aggiuntiva: la funzione di Eulero di *n*.

Vediamo prima qual è il procedimento da seguire per risalire al messaggio in chiaro e successivamente spiegheremo perché funziona.

Data $\varphi(n) = \varphi(p)\varphi(q) = (p-1)(q-1),$ determiniamo d soluzione della congruenza

$$ed \equiv 1 \mod \varphi(n)$$

Sappiamo che la congruenza ammette un'unica soluzione perché $(e, \varphi(n)) = 1$ e questo ci spiega anche perché nella scelta iniziale della chiave abbiamo richiesto che fosse $MCD(e, p-1) = 1$ e $MCD(e, q-1) = 1$.

Nel nostro caso, in cui la chiave pubblica è *(3, 1003)*, e la nostra informazione "privata" è $p=17$ e $q=59$ e quindi abbiamo che

$$\varphi(1003) = \varphi(17)\varphi(59) = 16 \cdot 58 = 928$$

Allora la soluzione della congruenza $3x \equiv 1 \mod (928)$ ci da la nostra chiave privata d =619.

A questo punto eleviamo ciascuno dei C_i ricevuti a d e riduciamo i valori ottenuti modulo n e ritroviamo esattamente i P_i di partenza, ovvero i blocchi del messaggio in chiaro che può così essere facilmente ricostruito.
Nell'effettuare quest'ultimo calcolo utilizziamo il metodo basato sulla numerazione binaria.
Dobbiamo quindi calcolare 328^{619}, 797^{619}, 825^{619}, 131^{619} e ridurli modulo 1003: $P_1 = 11,\ P_2 = 604,\ P_3 = 117,\ P_4 = 23,$ a cui però vanno aggiunti davanti tanti zeri quanti ne servono per rendere i blocchi ottenuti di 4 cifre, ossia: 0011, 0604, 0117, 0023.

Poiché ogni lettera corrisponde a 2 cifre (a partire da 00 che è la A) scopriamo la parola cifrata:

00	*11*	*06*	*04*	*01*	*17*	*00*	*23*
A	*L*	*G*	*E*	*B*	*R*	*A*	*X*

Abbiamo quindi scoperto che $P_i \equiv C_i^d \mod (n)$ e che quindi la nostra "chiave segreta" d ci consente di decifrare il messaggio.

Ma perché questo procedimento ha funzionato? E' facile dimostrarlo sfruttando i risultati che abbiamo imparato sulle congruenze:

$$C_i^d \equiv (P_i^e)^d = P_i^{e\,d} \mod (n)$$

Ma d'altra parte $ed \equiv 1 \mod \varphi(n)$ e quindi, per definizione di congruenza, $ed - 1$ è un multiplo di $\varphi(n)$, cioè

$$ed = 1 + k\ \varphi(n) \qquad \textit{per un certo } k$$

Quindi

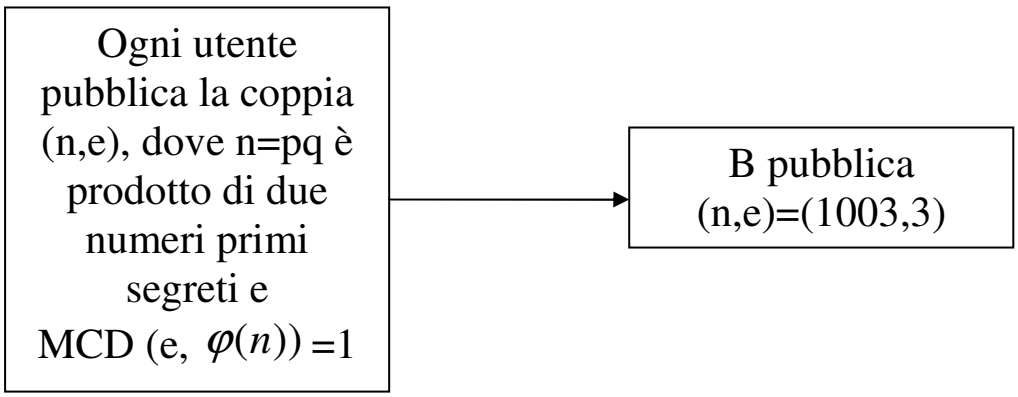
$$P_i^{ed} = P_i^{1+k\varphi(n)} = P_i (P_i^{\varphi(n)})^k$$

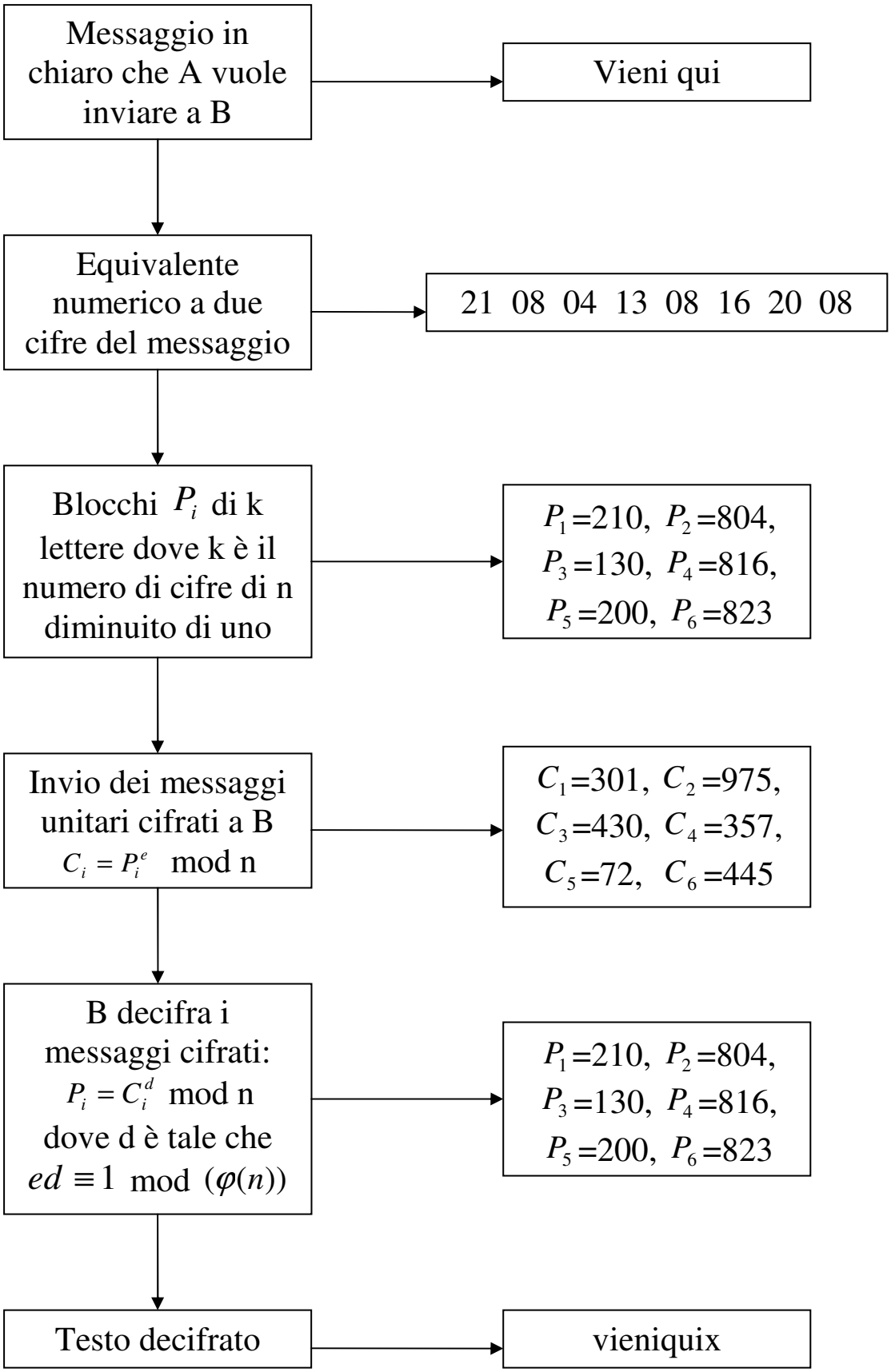
Poiché $MCD(P_i, n) = 1$, per il teorema di Eulero $P_i^{\varphi(n)} \equiv 1 \mod (n)$
e quindi

$$P_i^{ed} \equiv P_i \mod (n).$$

Osserviamo, infine, che quando il mittente ha cifrato il messaggio abbiamo richiesto che ciascun blocco di numeri P_i fosse minore di *n:* questo per garantire che non ci fosse ambiguità in fase di decrifratura nella determinazione del numero congruo a C_i^d modulo *n.*
Quest'ultimo è, infatti, l'unico numero che soddisfa la congruenza compreso tra 0 e *n-1.*

Ricostruiamo, per finire, tutto il viaggio da A a B del nostro messaggio in una tabella:




Messaggio in chiaro che A vuole inviare a B
Vieni qui
Equivalente numerico a due cifre del messaggio
21 08 04 13 08 16 20 08
Blocchi P_i di k lettere dove k è il numero di cifre di n diminuito di uno
P_1=210, P_2=804,
P_3=130, P_4=816,
P_5=200, P_6=823
Invio dei messaggi unitari cifrati a B
$C_i = P_i^e \mod n$
C_1=301, C_2=975,
C_3=430, C_4=357,
C_5=72, C_6=445
B decifra i messaggi cifrati:
$P_i = C_i^d \mod n$
dove d è tale che
$ed \equiv 1 \mod (\varphi(n))$
P_1=210, P_2=804,
P_3=130, P_4=816,
P_5=200, P_6=823
Testo decifrato
vieniquix

Settima lezione

Come abbiamo visto nel sistema di cifratura RSA, l'innovazione della chiave pubblica funziona per il fatto che solo il proprietario di tale chiave conosce la fattorizzazione di uno dei due numeri.
Tutto ciò che è stato difficilmente creato potrebbe quindi cadere facilmente se qualcuno riuscisse a scomporre in fattori qualsiasi numero.
Come abbiamo imparato già alle scuole inferiori ogni numero naturale lo possiamo scomporre come prodotto di numeri primi.
I numeri primi sono quei numeri, maggiori di 1, che sono divisibili solo per se stessi e per uno, e sono ad esempio: 2, 3, 5, 7… e molti altri, ma quanti altri?
Per ora ci preoccuperemo di mostrare che ogni numero è fattorizzabile in numeri primi.

Teorema fondamentale dell'aritmetica Sia n un numero intero maggiore di 1, allora:

$n=p_1^{H1} \times p_2^{H2} \times \ldots \times p_S^{HS}$

Dove $1 < p_1 < p_2 < \ldots < p_s$, sono tutti numeri primi distinti e i loro esponenti sono positivi, e tale decomposizione in fattori primi di n è unica.

Dimostrazione.
Esistenza: per induzione, se n=2 allora non c'è nulla da dimostrare perché 2 è numero primo.
Sia $2 < k < n$, e supponiamo di aver dimostrato per tutti quei k che esiste al fattorizzazione., e vediamo che ciò implica l'esistenza della fattorizzazione per n.
Se n è un numero primo, allora non c'è nulla da dimostrare.
Se n non è un numero primo allora lo possiamo scomporre come prodotto di due numeri a,b maggiori di 1 e minori di n: $n = ab$.
Quindi i numeri a , b appartengono a quell'insieme di numeri k di cui abbiamo supposto conoscere la fattorizzazione.
Possiamo quindi scrivere che le due fattorizzazione sono:

$$a = p_1^{h1} p_2^{h2} \cdots p_r^{hr} \quad \text{e} \quad b = q_1^{h1} \cdots q_s^{hs}$$

dove p_i e q_i sono tutti numeri primi distinti e gli esponenti sono tutti positivi.
Quindi ora possiamo scrivere che :

$$n = ab = p_1^{h1} p_2^{h2} \cdots p_r^{hr} q_1^{h1} \cdots q_s^{hs}$$

Quindi ora basta solo controllare se qualche numero p della fattorizzazione di a è uguale a qualche numero q della fattorizzazione di b e sommare i relativi esponenti, e così abbiamo ottenuto la fattorizzazione di n in numeri primi.

Unicità: per dimostrare che la fattorizzazione è unica, supponiamo per assurdo che ne esistano due dello stesso numero n:

$$n = p_1^{h1} \cdots p_r^{hr} = q_1^{k1} \cdots q_s^{ks}$$

Con tutti i numeri p, q k e h che rispettano le ipotesi precedenti, e con $r \neq s$.
Siccome il numero p_1 divide n, allora deve dividere anche la fattorizzazione di n scomposto in fattori q.
Sia p_1 che tutti i q_i sono numeri primi allora p_1 per dividere tale prodotto deve dividere un certo q_j, visto che sia p_i che q_i sono numeri primi e uno divide l'altro allora $p_i = q_i$.

Questo ragionamento lo possiamo fare per ogni p_i, quindi i p_i sono almeno quanto i q_i, ma lo stesso ragionamento vale anche per i q_i e quindi il numero dei numeri primi delle due differenti è lo stesso, cioè r = s e quindi:

$$n = p_1^{h1} \cdots p_r^{hr} = p_1^{k1} \cdots p_s^{ks}$$

Quindi per ora abbiamo dimostrato che le due fattorizzazione sono composte dagli stessi numeri primi, ora andiamo a considerare le differenti potenze dello stesso numero primo.
Se dividiamo entrambi i membri dell' uguaglianza per il primo numero primo p_1 elevato alla sua potenza h_1 otteniamo:

$$p_2^{h2} \cdots p_r^{hr} = p_1^{k1-h1} \cdots p_s^{ks}$$

Ma ora nel primo membro non compare nessuna potenza del numero p_1, oppure il numero p_1 compare con una potenza uguale a 0 , quindi anche nel secondo membro dell'equazione deve apparire con lo stesso esponente. Da ciò si deduce che $k_1 - h_1 = 0$ e quindi $k_1 = h_1$.
Ripetendo ora lo stesso procedimento per tutte le potenze otteniamo quindi che anche loro sono tutte uguali.
Abbiamo visto che se proviamo a scrivere un numero con due differenti fattorizzazioni, in realtà esse sono uguali.

ESERCIZI

Trovare al fattorizzazione dei seguenti numeri: 13, 48, 144, 98000.

Tutti i numeri primi sono numeri naturali, quindi sono un loro sottoinsieme, ma i numeri naturali sono infiniti, quanti sono quindi i numeri primi?
Come della scomposizione in fattori primi, anche di vedere quanti sono i numeri primi se ne è occupato già Euclide ne "Gli elementi".

Teorema I numeri primi sono infiniti.

Dimostrazione.
Per assurdo supponiamo che essi siano finiti, e di averli trovati tutti dal primo P_1 all'ultimo P_n.
Consideriamo ora un altro numero, ottenuto dal prodotto di tutti i numeri primi più 1 :

$a = p_1 \times p_2 \times \ldots \times p_n + 1$

Come è evidente questo numero è più grande dell'ultimo numero primo che abbiamo supposto conoscere.
Siccome abbiamo supposto che i numeri primi fossero finiti allora il nuovo numero non potrà essere un primo. Quindi a sarà sicuramente divisibile per uno dei numeri primi conosciuti, e fattorizzabile con alcuni dei numeri primi noti.
Come è noto se p divide n e n = a + b allora p divide sia a che b, e se p divide sia a che b allora p divide n.
Supponiamo che nella fattorizzazione del numero a compaia il primo p.
Tale numero p deve allora dividere $p_1 \times p_2 \times \ldots \times p_n + 1$, cioè deve dividere sia il prodotto di tutti i numeri primi sia il numero 1.
Sicuramente dividerà il prodotto di tutti i numeri, e ciò è vero perché nel prodotto compare anche il numero p.
Sicuramente non dividerà il numero 1 , perché p è maggiore di 1.

Siccome p non divide entrambi i termini della somma allora p non divide a, e tale ragionamento vale per tutti i numeri primi che abbiamo supposto di conoscere.

Il numero a non è allora divisibile per nessun numero primo, abbiamo quindi due possibilità:
1) Il numero a è un numero primo
2)Esiste un numero primo p che è diverso da tutti i numeri primi che abbiamo prima detto essere finiti.

In entrambi i casi abbiamo un nuovo numero primo che è diverso dai precedenti.
Quindi per quanti primi possiamo conoscere, ne possiamo trovare sempre uno diverso, ossia i numeri primi sono infiniti.

ESERCIZI
Verificare che se *a* è della forma $p_1 \times p_2 \times \ldots \times p_n + 1$ con tutti i p_i numeri primi allora *a* non è divisibile per alcun numero p_i.

- $2 \times 3 \times 5 + 1 =$
- $2 \times 3 \times 5 \times 7 + 1 =$
- $2 \times 3 \times 5 \times 7 \times 11 \times 13 \times 17 + 1 =$

Come abbiamo imparato sin dalle scuole elementari, per calcolare il massimo comun divisore o il minimo comune multiplo tra due numeri bisogna procurarsi la fattorizzazione dei due numeri dati e poi prendere i primi con le relative esponenti che ci interessano.
Tutto si riduce quindi a scomporre un numero in fattori, e come abbiamo visto in precedenza il sistema RSA si basa proprio sulla difficoltà di fattorizzare numeri grandi.
Vediamo quali problemi si pongono nel fattorizzare un numero.
Come prima cosa bisogna vedere quali numeri primi dividono tale numero, e poi preoccuparsi con quale potenza essi compaiono nella fattorizzazione.
Per numeri piccoli possiamo procedere per tentativi, e poi esistono dei piccoli trucchi.
Ad esempio se vogliamo sapere se il numero 2 , che è un numero primo, divide un certo numero, sappiamo bene che è sufficiente controllare solo se l'ultima cifra di tale numero è pari o dispari.
Se volessimo controllare se un numero è divisibile per 3, dobbiamo controllare se dalla somma delle sue cifre si ottengono i numeri 3, 6 o 9 che sono tutti divisibili per 3.
Un altro esempio è del primo 5, che divide un numero se e solo se l'ultima cifra di tale numero è 0 o 5.
Esistono altri criteri per altri numeri, come 7 o 11, ma di sicuro non esistono criteri per tutti i numeri

Proposizione Un numero è divisibile per 2 se la sua ultima cifra è divisibile per 2.

Dimostrazione.
Siccome contiamo in base 10 allora il numero composto dalle cifre *abc* è il numero ottenuto da:
$a \times 100 + b \times 10 + c \times 1$ (ad esempio $123 = 1 \times 100 + 2 \times 10 + 3 \times 1 = 100 + 20 + 3$).
Dalla teoria delle congruenze studiata precedentemente sappiamo che se un numero è divisibile per 2 allora tale numero è congruo 0 modulo 2.
Essendo $10 = 5 \times 2$ allora di sicuro 10 è congruo 0 modulo 2, e lo stesso vale per tutte le potenze di 10, e per i loro multipli.
Quindi il numero $a \times 100 + b \times 10 + c \times 1$ è congruo a c modulo 2, e tale numero sarà divisibile per 2 se e solo se il numero c è divisibile per 2.

Proposizione Un numero è divisibile per 3 se la somma delle sue è divisibile per 3.

Dimostrazione.
Con lo stesso ragionamento di prima dobbiamo verificare se il numero
abc = a x 100 + b x 10 + c x 1 è divisibile per 3.
Il numero 10 = 9 + 1 quindi è congruo 1 modulo 3.
Il numero 100 = 9 x 11 + 1 che è congruo 1 modulo 1.
Così via per tutte le potenze di 10, quindi ogni potenza di 10 è congrua 1 modulo 3.
Il numero a x 100 + b x 10 + c x 1 sarà quindi congruo ad a + b + c modulo 3 e tale numero sarà divisibile per 3 se è uguale a 3 6 o 9(in caso sia ancora maggiore di 9 possiamo ripetere il procedimento).

Comunque ci viene dato un numero possiamo controllare se esso è divisibile per dei piccoli numeri primi, ma se così non fosse non ci resta che procedere per tentativi.
Dato il numero a , dobbiamo controllare se esso è divisibile per tutti i numeri primi compresi tra 2 e $a/2$ (se a è dispari basta $(a-1)/2$), perché tale numero primo deve essere moltiplicato almeno per 2 quindi non può essere più grande della metà di a.
Ad esempio se volessimo calcolare la fattorizzazione di 20420400 dovremmo controllare tutti i numeri primi compresi tra 2 e 10210200 che non sono poi pochi.
Dovremmo sederci muniti di carta e penna, pazienza e coraggio e cominciare:

1) 2 divide 20420400? (si perché l'ultima cifra è 0 che "possiamo considerare pari")

2) 3 divide divide 20420400? (si perché 2 + 0 + 4 + 2 + 0 + 4 + 0 + 0 = 12 = 2+1 =3)

3) 5 divide 20420400? (si perché l'ultima cifra è 0)

p) p divide 20420400?

Per farlo dobbiamo scrivere :

20420400 = p x q + r e se r = 0 allora p divide 20420400 altrimenti no .

E così via fino ad arrivare all'ultimo numero primo più piccolo di 10210200.
Ogni volta poi che troviamo un primo che divide il nostro numero dobbiamo anche verificare qual è la potenza massima con cui tale primo divide il nostro numero.

ESERCIZI
Trovare la fattorizzazione dei seguenti numeri:
34, 21000, 101, 1234567.

E' evidente come tale procedimento, se applicato a numeri che si usano nella chiave pubblica dell' RSA , che sono molto più grandi delle sole 8 cifre di 20420400, è praticamente inapplicabile per poter fattorizzare numeri "enormi".
Logicamente però grazie alla tecnologia possiamo usare pc, programmi avanzati, e invece che usare carta e penna, utilizzare la memoria del nostro pc.
Ma anche questo tentativo è vano, i numeri che vengono presi in considerazione sono troppo grandi perché chi pubblica una chiave si preoccupi che qualcuno riesca a violare la sua segretezza.
Abbiamo visto come fattorizzare un numero qualsiasi, ma come facciamo a saper se un dato numero è primo o no? Cioè se è utile cercarne una fattorizzazione o meno.

Ci sono due possibilità, possiamo banalmente cercarne la fattorizzazione, e se essa non esiste allora tale numero è primo.
Ad esempio, se cerchiamo la fattorizzazione di 13, ci rendiamo conto che 13 non è divisibile per 2, 3 o 5 (7 > (13-1)/2 quindi è inutile controllare se è un divisore di 13) e quindi 13 è un numero primo.
Oppure possiamo utilizzare qualche test di primalità.

Il più semplice test di primalità è il *crivello di Eratostene*.
Si decide tra quale intervallo di numeri si vogliono cercare i numeri primi, ad esempio tra 1 e 100.
Si parte dal "primo numero primo" ossia 2,e si eliminano tutti i numeri divisibili per 2.
Successivamente si considera il primo numero non cancellato che sarà un numero primo, cioè 3, e si eliminano tutti i numeri divisibili per 3 tra tutti quelli rimasti(ad esempio 6 è già stato cancellato).
Continuiamo il procedimento per 5, per 7 e così via.
Fino a che non rimangono tutti numeri che sono primi e quindi non ci resta più nulla da cancellare.
In questo modo calcolare i numeri primi tra 1 e 100 non è poi molto difficile, ma se amentiamo il nostro intervallo di numeri il tempo impiegato aumenta.

Un altro test di primalità è quello presentato da *Wilson*.
Un numero n è primo se e solo se (n-1)! +1 è divisibile per n.
Ricordiamo che n fattoriale è n! = n x (n-1) x (n-2) x …x 2 x 1.
Ossia n è primo se e solo se (n-1)! È congruo -1 modulo n.
Siccome il calcolo del fattoriale aumenta di difficoltà molto velocemente (provate a far calcolare alla calcolatrice 80!) e quindi anche questo test non è molto applicabile, perché con numeri non troppo grandi diventa subito inapplicabile.

ESERCIZI
Verificare con il test di Wilson che 3, 5 e 7 sono numeri primi.

Un test simile è quello che si ottiene dal piccolo teorema di Fermat che afferma :
se p è primo allora comunque si sceglie a si avrà che a^p è congruo ad a modulo p.
Quindi con questo teorema otteniamo un test di "non – primalità", cioè se esiste un numero a tale che a^p non è congruo ad a modulo p allora p non è primo.
Ad esempio possiamo verificare che 10 non è primo usando come a il numero 2.

Il *test di Lucas* afferma che:
Sia n> 1, se esiste un intero a tale che a^{n-1} è congruo a 1 modulo n e a^m non è congruo 1 modulo n per ogni 1 < m < n-2 allora n è primo .

Tutti questi test oltre ad avere un difficile dimostrazione sono anche piuttosto lunghi da verificare, quindi di nuovo sembra che il sistema RSA possa considerarsi per ora un ottimo sistema di crittografia.
La ricerca di nuovi numeri primi, o della fattorizzazione dei numeri è un campo ancora aperto, e ben ripagato. Quindi a quanto pare lo studio della matematica, oltre alla sopravvivenza scolastica, potrebbe permettere anche un buon guadagno si denaro, anche se in maniera non poco faticosa.

Ottava lezione

Quest'ultima lezione si svolge nel laboratorio di informatica. Sui computer verrà installato il software Maple, un programma matematico che permette di effettuare molte interessanti sperimentazioni. Maple, infatti, è in grado, attraverso semplici istruzioni, di risolvere equazioni e sistemi, semplificare espressioni, calcolare derivate, integrali, limiti, valutare o approssimare valori di espressioni o di funzioni in un particolare punto, ecc.
L'ambiente di lavoro è un semplice foglio su cui vengono digitati i comandi e all'interno del foglio stesso il software fornisce immediatamente le risposte. Le istruzioni sono davvero facilissime da imparare e non richiedono nessuna conoscenza o familiarità con i linguaggi di programmazione (almeno per quanto riguarda le applicazioni basilari).
Naturalmente in una sola lezione è impossibile imparare ad usare davvero il programma, anche in maniera elementare. Però è sufficiente per lo scopo che ci prefiggiamo: toccare "con mano" la difficoltà della fattorizzazione almeno in termini di tempo necessario.

L'idea è quella di utilizzare il comando **ifactor(n)**, dove *n* è l'intero da fattorizzare, per verificare come fino ad un certo ordine di cifre (ad esempio abbiamo provato con 25/30 circa) la risposta del calcolatore sia immediata o comunque dell'ordine di qualche decimo di secondo, mentre per numeri con tante cifre (come quelli usati nell'RSA) il tempo del calcolatore aumenti sensibilmente.
Maple è dotato di una istruzione, **time()**, che, effettuata dopo ogni operazione, ci restituisce proprio il tempo di risposta del calcolatore. Mostriamo un esempio:

```
> ifactor(25);
                                   (5)^2

> time();
                                   0.110

> ifactor(89326);
                                (2) (59) (757)

> time();
                                   0.126

> ifactor(74023819593715058739728948);
                    (2)^2 (3)^2 (23) (387784664719) (702937) (32797)

> time();
                                   0.141

> ifactor(375027491640322886792855949309238647839200l);
          (19) (249229639642279) (1952857271238834007) (405545243)

> time();
                                   15.063

> ifactor(75037729587295017739462285912874098765394758993291);
     (26215764034350799335784261344 61) (815103613) (35115943987)

> time();
                                   55.125

> ifactor(752972745436870989765432678796554689075425876598788653478900987665);
     (3) (5) (13) (411357811644598812397146142918734353295386201) (538953807820091) (17417)

> time();
                                   403.359

>
```

Nell’immagine abbiamo fattorizzato prima il numero 25 e il tempo impiegato è stato un decimo di secondo (ma in realtà anche meno, perché alcuni centesimi di secondo sono necessari al programma per avviare la finestra di lavoro e quindi il nostro cronometro non parte esattamente da zero).
Successivamente abbiamo fattorizzato un numero di 5 cifre e il calcolo ha richiesto solo 1 centesimo di secondo (la funzione **time** opera come un cronometro che non si azzera dopo ogni operazione, ma va avanti per tutto il tempo in cui è attivo il foglio di lavoro; quindi è come se dopo ogni fattorizzazione stoppassimo il cronometro e lo riavviassimo dal punto in cui si era fermato nella fattorizzazione successiva). Passando poi a un numero di 25 cifre, osserviamo che il tempo di risposta è ancora praticamente immediato. Aumentando ancora le cifre a caso (e portandole a 45) il tempo aumenta in modo più sensibile: il calcolatore ha bisogno di quasi 15 secondi! Ma è negli ultimi due passaggi dell’immagine illustrativa che osserviamo come il tempo di risposta cresca in maniera sempre più decisa: fattorizzare un numero a caso di 50 cifre ha richiesto ben 40 secondi; di 65 addirittura più di 6 minuti!
Naturalmente siamo ancora lontani dai numeri e dai tempi coinvolti nell’RSA, ma questo piccolo “esperimento” ci da comunque un esempio pratico del fatto che quanto detto nel corso delle lezioni teoriche ha una solida base di verità.

I comandi **prevprime(n)** e **nextprime(n)** ci restituiscono rispettivamente il numero primo precedente e successivo rispetto a *n*. Anche queste istruzioni ci suggeriscono degli utili esercizi: fornendo, infatti, come dato *n* un intero arbitrariamente grande, i comandi ci restituiscono entrambi un numero primo su quell’ordine di grandezza. E’ utile, quindi, per due finalità:

1) intuire perché l’RSA funziona, in quanto si vede con un semplice confronto come siano sensibilmente diversi i tempi per generare un primo e per fattorizzare un numero dello stesso ordine di grandezza. Ad esempio:

```
> prevprime(75297274543687098976543267879655468907542587659878865347890098 7665);
                              752972745436870989765432678796554689075425876598788653478900987481
> time();
                                                  0.126
> nextprime(752972745436870989765432678796554689075425876598788653478900987665);
                              752972745436870989765432678796554689075425876598788653478900987681
> time();
                                                  0.142
> ifactor(752972745436870989765432678796554689075425876598788653478900987665);
          (3) (5) (13) (411357811644598812397146142918734353295386201) (538953807820091) (17417)
> time();
                                                  391.092
```

2) simulare (unendo insieme i comandi next/prevprime e ifactor) come opera l'RSA: si possono produrre in pochi secondi due primi arbitrariamente grandi e moltiplicarli (generando, così, la nostra chiave) e poi verificare successivamente quanto il calcolatore impiega a fattorizzare il numero trovato. Certo, ovviamente non possiamo arrivare a cifre dell'ordine di quelle realmente usate, ma come simulazione è efficace:

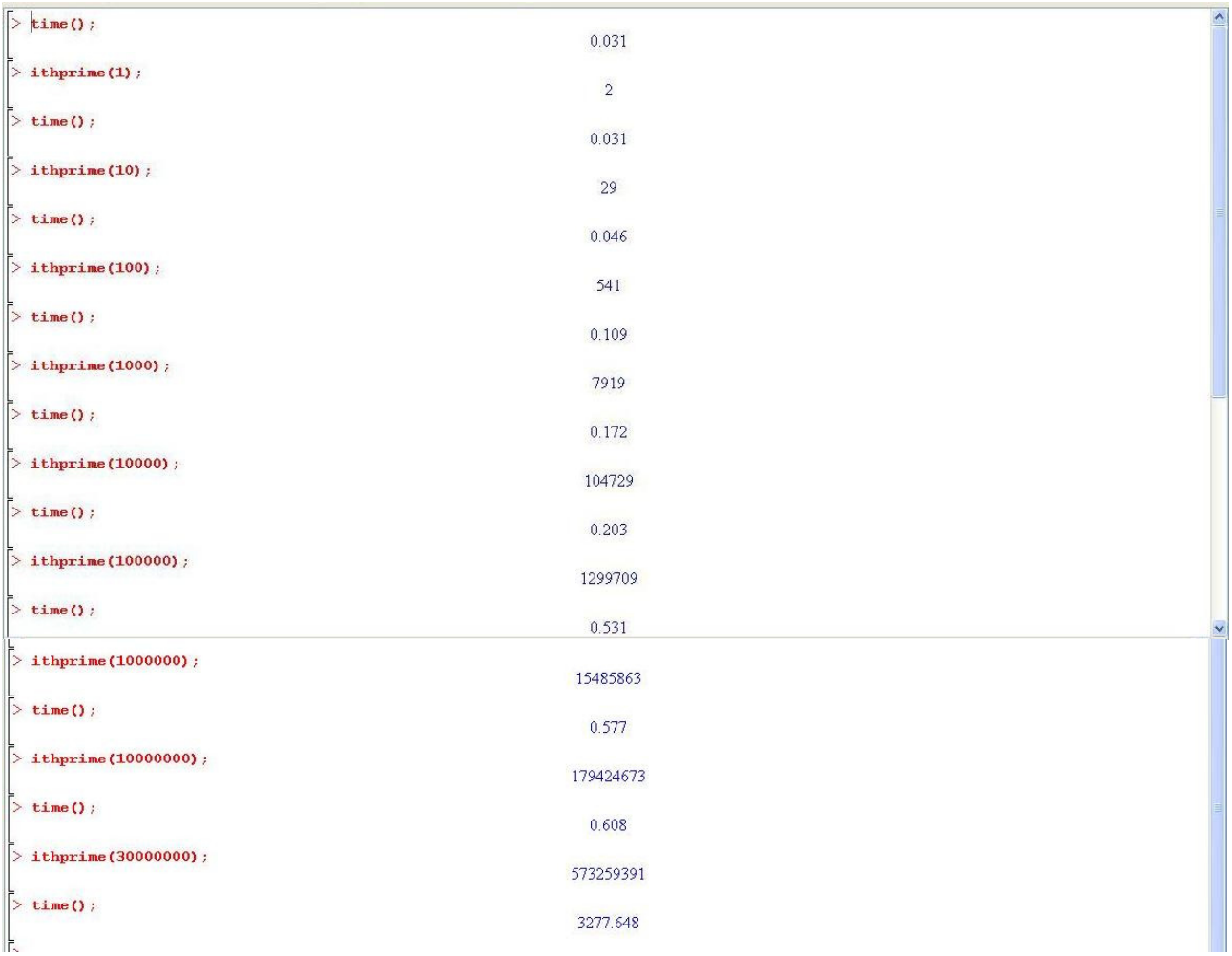
```
> prevprime(63826384783927849376);
                                        63826384783927849367
> time();
                                        0.063
> nextprime(73982759849365483527);
                                        73982759849365483531
> time();
                                        0.142
> 63826384783927849367*73982759849365483531;
                                        4722052097522529327143602288018887274877
> time();
                                        0.268
> ifactor(4722052097522529327143602288018887274877);
                                        (73982759849365483531) (63826384783927849367)
> time();
                                        122.302
```

Un'altra possibile esercitazione si basa sul comando **ithprime(n)**, con *n* intero, che ci fornisce l'*n*-esimo numero primo. Nella lezione introduttiva avevamo accennato al fatto che la teoria dei numeri primi è ancora parzialmente sconosciuta ai matematici: c'eravamo lasciati con il premio in denaro messo in palio per la dimostrazione della congettura di Riemann, che tentava di dare una struttura ai numeri primi.

Questo esercizio ci da una piccola idea di come questa struttura attualmente manchi: se chiediamo a un calcolatore di generare un numero primo arbitrariamente grande, questa operazione viene effettuata in pochissimi decimi di secondo (come abbiamo visto), ma se chiediamo di restituirci un numero primo in una specifica posizione (ad esempio il 30.000.000esimo primo), il computer impiega abbastanza, perché è costretto ad effettuare test su tutti i numeri finché non trova il 30.000.000esimo primo! Non è in grado, cioè, di conoscerlo a priori.

In pratica, se partiamo da 2 (che è il primo) e a seguire moltiplichiamo il numero delle cifre di *n* per 10, il tempo aumenta di pochi centesimi o decimi di secondo. Ma la vera "svolta" in negativo (e soprattutto catastrofica) si ha oltre le 10.000.000 di cifre: per calcolare il 30.000.000esimo primo, il tempo richiesto è di quasi un'ora!!!

Ecco un esempio:



Per completezza, diciamo che esiste anche un'istruzione basata sui testi di primalità; **isprime(n)**, che ci restituisce *true* se l'intero *n* è primo e *false* se non lo è (anche se non con assoluta certezza, infatti il programma esegue un test di Lucas).

ESEMPI:

```
> isprime(1);
```
false

```
> isprime(2);
```
true

```
> isprime(17);
```
true

```
> isprime(21);
```
false

```
> isprime(11!+1);
```
true

```
> isprime(2^30*3^30+7);
```
true

Se l'argomento risulta interessante (e soprattutto se rimane tempo) si può insegnare l'uso di quasi tutti i comandi basilari di Maple (quelli che non richiedono costruzioni sintattiche):

- **Solve(eqn,var)** per risolvere equazioni o **solve(eqns,vars)** per i sistemi;
- **fsolve(eqns,vars,options)** per soluzioni numeriche di equazioni;
- **diff(f(x),x),**
- **int(expr, x),**
- **limit(f(x), x=a)** (queste ultime tre se hanno già studiato derivate, integrali e limiti);
- **plot(f,h,v)** per disegnare grafici di funzioni in un certo rettangolo (dato dai parametri h e v)
- **plot3d(expr, x=a..b,y=c..d)** per grafici tridimensionali

ecc.

ESEMPI

```
> solve( x^2+x=1, x );
```
$$-\frac{1}{2}+\frac{\sqrt{5}}{2}, -\frac{1}{2}-\frac{\sqrt{5}}{2}$$

```
> solve({u+v+w=1, 3*u+v=3, u-2*v-w=0});
```
$$\left\{ v=\frac{3}{5}, w=\frac{-2}{5}, u=\frac{4}{5} \right\}$$

```
> fsolve( tan(sin(x)) = 1, x );
```
0.9033391108

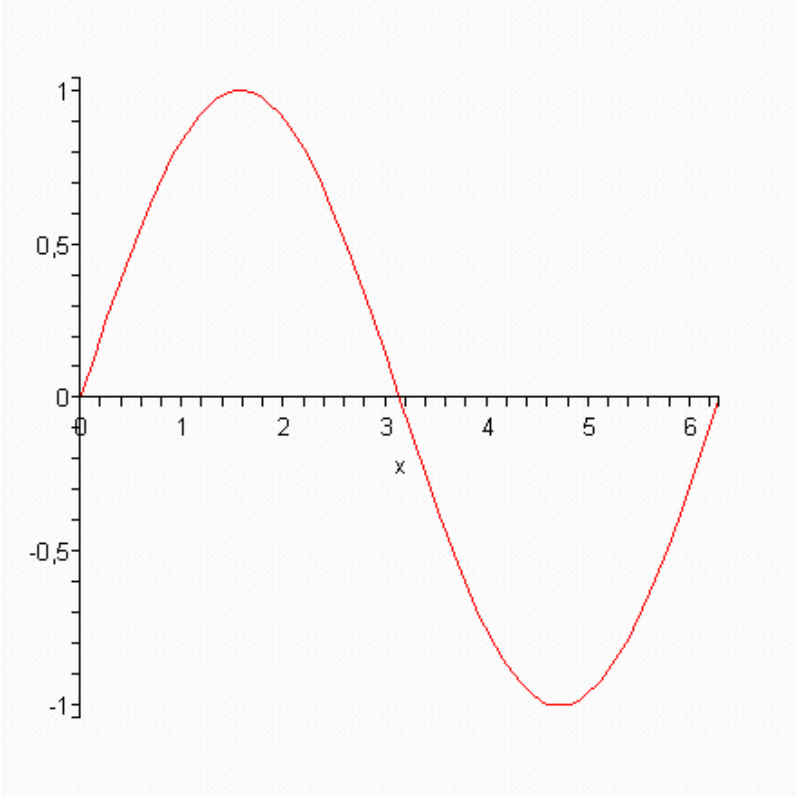
```
> diff( x^2+x^4-3*x+2, x );
```
$2x+4x^3-3$

```
> int( sin(x), x );
```
$-\cos(x)$

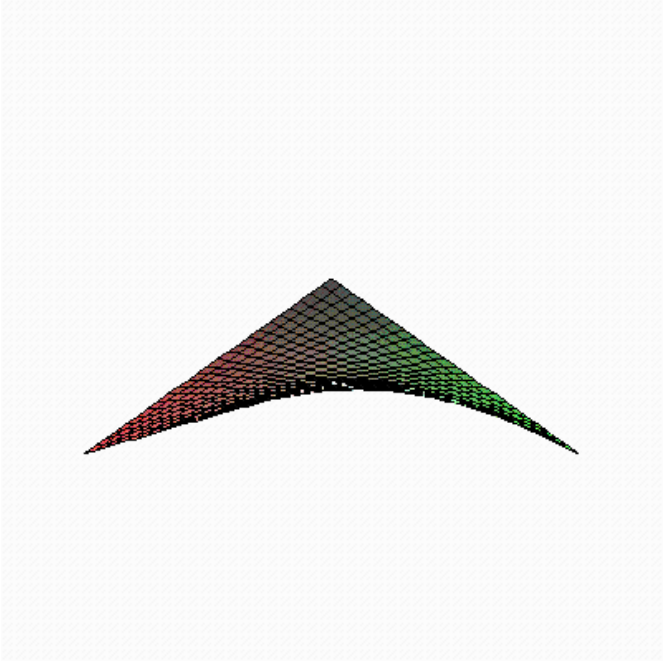
```
> limit( sin(x) / x, x = 0 );
```

1

```
> plot(sin(x), x = 0..2*Pi);
```




```
> plot3d(sin(x)*y, x = 0..1,y=0..1);
```



Bibliografia

G.M. Piacentini Cattaneo, *Algebra. Un approccio algoritmico,* Decibel-Zanichelli, Padova, 1999.

M.W. Baldoni, C. Ciliberto, G.M. Cattaneo Piacentini con la collaborazione di A. Calabri, *note di Matematica discreta,* Aracne, Roma, 2002.

Simon Singh, *Codici & segreti,* Fabbri Editori, Milano, 2002.

Marcus Du Satoy, *L'enigma dei numeri primi – l'ipotesi di Riemann, l'ultimo grande mistero della matematica,* Rizzoli, Milano, 2004.

Laura Catastini, *Il pensiero allo specchio,* La Nuova Italia, Scandicci (Firenze), 1990.