Parte V

Condividere segreti e combattere il crimine — *la Crittografia*

Condividere segreti e combattere il crimine

Sicuramente avrete sentito parlare di spie e agenti segreti che usano codici cifrati e magiche scritture invisibili per scambiare messaggi. È per questo che è nata la "crittografia", l'arte di scrivere e di decifrare codici segreti. Durante la seconda guerra mondiale, gli inglesi crearono una macchina specificatamente disegnata per decifrare i codici militari. Poi venne l'era dei computer e cambiò ogni cosa e la crittografia entrò in una nuova era. Una quantità enorme di calcoli, inimmaginabile precedentemente, potè essere utilizzata per decifrare i codici. Quando le persone iniziarono a condividere l'uso di computer fra loro, le parole d'ordine o *password* iniziarono ad essere utilizzate non per entrare in una base militare ma per proteggere la riservatezza delle informazioni. Quando poi i computer vennero interconnessi da reti, ci furono ulteriori motivi per proteggere le informazioni da quanti avrebbero voluto leggerle in modo non autorizzato. Con la posta elettronica poi si pose il problema di certificare che chi firma un messaggio sia veramente chi dice di essere. Ora le persone fanno transazioni bancarie *on-line*, tramite servizi Web, sulla rete Internet. Oggi si acquistano e si vendono merci usando i computer. E' necessario per questo che gli strumenti usati per spedire gli ordini e per trasferire il denaro siano sicuri. Il crescente pericolo di un attacco terroristico tramite l'uso di computer rende il problema della sicurezza informatica sempre più importante.

La crittografia probabilmente vi farà pensare a *password* segrete e a come miscugliare le lettere di un messaggio in modo che il nemico non possa leggerlo. La realtà è molto differente. I sistemi moderni di elaborazione non memorizzano le *password* segrete, perché se lo facessero chiunque riuscisse ad averne accesso potrebbe violare completamente la sicurezza del sistema. Questo sarebbe disastroso: potrebbero fare false operazioni bancarie, spedire messaggi fingendo di essere qualcun altro, leggere file segreti, dare ordini ad eserciti, far cadere governi. Oggi le password sono gestite tramite le "funzioni a senso unico" (*one-way function*) di cui abbiamo parlato nell'attività 15. E per criptare i messaggi non si mischiano semplicemente le lettere del messaggio: si usano tecniche basate su problemi molto difficili da risolvere, i problemi "intrattabili" che abbiamo introdotto nella parte IV.

In questa sezione scopriremo un modo semplice per calcolare l'età media delle persone di un gruppo senza che nessuno del gruppo debba dichiarare la propria. Imparerete come due persone che non si fidano l'una dell'altra, possano giocare a testa o croce ed essere certe della correttezza del risultato sebbene abitino in città diverse e quindi non possano usare la classica monetina. Scoprirete poi un metodo per codificare messaggi segreti in modo che una sola persona possa

decodificarli anche se il metodo di codifica è noto a tutti.

Per gli insegnanti

Le attività che seguono forniscono allo studente una esperienza pratica sulle moderne tecniche di crittografia digitale, che sono molto differenti da quanto si sente dire comunemente in giro. Ci sono innanzitutto due concetti chiave. Il primo è la nozione di "protocollo", che è una definizione formale di una transazione. Il protocollo può farci ricordar il lavoro dei diplomatici, che devono rispettare una certa *etiquette* nelle loro parole ed azioni. Anche i computer hanno le loro regole di *etiquette* da rispettare! Compiti che appaiono difficilissimi possono essere portati a compimento da protocolli sorprendentemente semplici. L'attività 17, che può essere svolta in pochi minuti, mostra come un insieme di persone possano, di comune accordo, calcolare l'età (o il reddito) medio del gruppo senza che alcuno di loro debba dichiarare ad altri la propria età (o il proprio reddito). Il secondo concetto è che la complessità computazionale, l'intrattabilità, può ricoprire un ruolo fondamentale per proteggere il dialogo con altri attraverso l'uso di computer. L'attività 18 mostra come due persone, che non necessariamente si fidino l'una dell'altra, possano concordare sul risultato di una partita a testa o croce sebbene possano parlare solo al telefono (questa attività introduce anche i concetti di algebra di Boole e di circuiti logici, che possono essere approfonditi in altri percorsi didattici). L'attività 19, infine, mostra come le tecniche crittografiche consentano di criptare messaggi in modo sicuro anche se il metodo usato è conosciuto pubblicamente.

Alcune di queste attività, in modo particolare l'ultima, richiedono uno sforzo non banale. È necessario che motiviate la vostra classe giocando sul senso di meraviglia che accompagna la risoluzione di problemi che sembrano impossibili. È necessario creare questo stupore e fermarsi ripetutamente durante l'esposizione dell'attività per fare in modo che gli studenti non perdano la visione della (affascinante) luna perché osservano solo il (faticoso) dito. Queste attività sono fra le sfide più ardite di questo libro e sono anche le più tecnicamente intricate. Se ritenete che siano troppo complesse, potete saltare direttamente alla successiva parte VI, che ha una natura completamente differente, non tecnica.

Per le menti più tecno-curiose

Man mano che i computer invadono le nostre vite quotidiane, l'uso e le reali finalità della crittografia possono diventare non chiare. Molte persone non sono consapevoli delle capacità dei moderni metodi di crittografia. Il risultato è che quando grandi istituzioni, siano esse pubbliche o private, configurano sistemi che elaborano informazioni personali, tendono a delegare a tecnocrati le scelte chiave su come le informazioni verranno richieste, su quali verranno elaborate, su quali verranno fornite e a chi. Se le persone comprendessero meglio le

possibilità consentite dalla tecnologia moderna, potrebbero partecipare più attivamente a queste decisioni e la società potrebbe avere una differente, più equa infrastruttura di gestione delle informazioni

Gli argomenti trattati in questa sezione del testo ovvero i protocolli di gestione dell'informazione nascosta, i protocolli crittografici, la crittografia a chiave pubblica, sono generalmente considerati contenuti avanzati dei corsi di informatica. Ma le idee alla base non sono difficili da comprendere. Sono gli aspetti tecnici, non i concetti sottostanti, ad essere complessi. Nelle situazioni pratiche relative all'e-commence, per esempio, gli aspetti tecnici sono nascosti all'interno del software che rende molto semplice l'uso di questi strumenti. Ma è molto importante comprendere le idee che consentono a questi strumenti di funzionare, per poter aver la consapevolezza di cosa possa essere fatto e cosa no.

I sistemi crittografici sono di grande interesse per i governi, non solo perché vogliono mantenere la sicurezza delle loro comunicazioni ufficiali, ma anche per la preoccupazione che la comunicazione criptata possa essere usata da persone coinvolte in attività illegali come il traffico di droga o il terrorismo. Le intercettazioni delle comunicazioni di queste persone possono diventare inutili se non c'è modo di decodificare i loro messaggi. Queste preoccupazioni hanno creato un vasto dibattito fra chi deve mantenere la legalità (che vuole limitare la inviolabilità dei sistemi crittografici) e chi difende le libertà civili (che non accetta che il governo possa aver accesso alle comunicazioni private). Il governo degli Stati Uniti per un certo periodo impose restrizioni all'uso di alcuni metodi crittografici assimilandoli ad armi, come se fossero bombe o fucili. Chiunque può creare un canale di comunicazione sicuro, a patto che abbia le informazioni corrette e la capacità tecnica, ma questi strumenti possono essere pericolosi se posti nelle mani sbagliate. Ad un certo punto ci fu anche un esteso dibatto in merito al "Clipper Chip", un circuito integrato per la crittografia che avrebbe dovuto gestire una password aggiuntiva chiamata *chiave di scorta*, Questa chiave, che doveva essere in possesso di un'agenzia del governo, consentiva di decriptare ogni comunicazione. L'FBI e il Ministero della giustizia volevano che questo chip venisse ampiamente usato nei dispositivi di comunicazione. Il 'Clipper Chip" è stato fortemente avversato perché viola la privacy dei cittadini. Ogni tipo di protocollo crittografico può essere tecnicamente costruibile, un altro discorso è se sia politicamente accettabile o meno.

Le idee della crittografia hanno molte applicazioni oltre alla trasmissione di messaggi segreti. Per esempio è possibile verificare che i messaggi siano stati effettivamente spediti da chi dice di averlo fatto: questa è la "autenticazione", senza la quale il commercio elettronico non potrebbe esistere. Ci sono metodi per fare sì che le persone possano votare senza che il loro voto possa essere visto da altri, neanche da chi gestisce il sistema di elaborazione, ma riuscendo in ogni modo a scrutinare i voti

in modo corretto. È anche possibile giocare a carte al telefono, che può sembrare inutile e sciocco fino a quando non si comprende che stringere accordi commerciali non è molto diverso dal giocare a poker.

Ma come è mai possibile mischiare un mazzo di carte e giocare via telefono in competizione con una persona all'altro capo della comunicazione e di cui non ci si può fidare? Come si può riconoscere se qualcuno ha intercettato un messaggio, l'ha modificato e l'ha consegnato al posto dell'originale? Se non si potessere fare queste cose non sarebbe possibile fare commercio elettronico. Occorre impedire a criminali esperti dal punto di vista tecnico di poter forgiare false autorizzazioni di prelievo da conti bancari a partire dai dati intercettati sulla linea che connette il terminale del bancomat alla banca. Occorre che ditte malintenzionate non possano tecnicamente far fallire un concorrente generando falsi ordini o falsi contratti. Con la crittografia moderna questi miracoli possono essere compiuti e le seguenti attività indicano come.

Ci sono molti libri interessanti sui codici crittografici e sulla loro violazione. *Codebreakers: the inside story of Bletchley Park* curato da Hinsley e Stripp, fornisce una descrizione diretta di come alcuni dei primi computer vennero utilizzati per decodificare i codici militari durante la seconda guerra mondiale, consentendo così la fine delle ostilità e salvando in questo modo molte vite umane.

Attività 17

Condividere i segreti — *Protocolli che nascondono l'informazione*

Sommario

I protocolli crittografici ci consentono di condividere informazioni con altre persone, mantenendo nel contempo un sorprendente livello di privacy. Questa attività illustra una situazione nella quale l'informazione viene condivisa sebbene non venga rivelata: un gruppo di studenti calcolerà la media delle loro età senza che alcuno di loro debba rivelare la propria.

Abilità

✓ Calcolare una media

✓ Generare numeri casuali

✓ Lavorare in gruppo

Età

✓ a partire dai 7 anni

Materiale

Ogni gruppo deve avere:

✓ un blocchetto di carta

✓ una penna.

Condividere i segreti

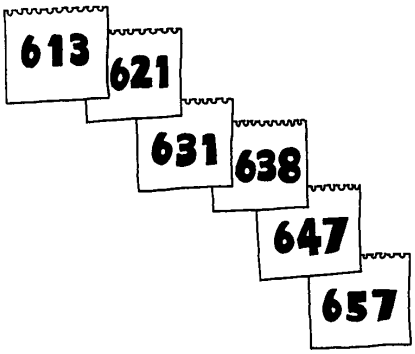
Introduzione

Scopo di questa attività è di calcolare la media delle età di un gruppo di studenti senza che ognuno di essi debba rivelare la propria. In alternativa con lo stesso metodo è possibile calcolare la media del reddito (della paghetta) degli studenti del gruppo o la media di qualche altro dato riservato. Questo metodo di calcolo funziona particolarmente bene con gli adulti, perché sono spesso particolarmente riservati su dati riservati come l'età o il reddito.

Il gruppo deve essere formato da almeno tre studenti.

Discussione

1. Spiegate al gruppo che volete conoscere l'età media dei partecipanti senza che ognuno debba dire agli altri la propria età. Chiedete suggerimenti su come pensano di fare e se pensano che questo problema ammetta una soluzione.
2. Selezionate da sei a dieci studenti (circa) per fare l'esperimento. Date il blocchetto al primo studente e chiedetegli di scrivere un numero casuale di tre cifre sul primo foglio. In questo esempio poniamo che lo studente abbia scelto 613.



3. A questo punto il primo studente deve strappare il primo foglietto dal blocco, aggiungere la propria eta' al numero casuale scelto e scrivere il risultato della somma nel secondo foglietto. Se lo studente ha otto anni il numero da scrivere nel secondo foglietto è 621. Lo studente deve tenere il foglio staccato dal blocco e non mostrarlo a nessuno.
4. Il blocchetto viene passato al secondo studente, che stacca dal blocco il foglio in cima, quello con il numero scritto dal collega,

calcola la somma fra tale numero e la propria età e scrive il totale sul foglietto successivo.

5. Occorre ora continuare il processo che vede ogni studente strappare un foglietto dal blocco, sommare la propria età e scrivere il risultato sul foglietto successivo. Il meccanismo continua fino a quando tutti gli studenti non hanno ricevuto il blocco.
6. A questo punto il blocco viene riconsegnato al primo studente che sottrae dal valore scritto sul blocco quello del numero casuale che aveva scelto all'inizio. Nell'esempio il blocco è stato fatto girare fra i cinque studenti del gruppo e al numero finale, 657, viene sottratto il numero casuale iniziale, 613. Il risultato è 44. Questa è la somma delle età degli studenti. La media a questo punto può essere calcolata in modo semplice dividendo questa somma per il numero degli studenti del gruppo. Nel nostro esempio l'età media risulta di 8.8 anni.
7. Sottolineate che, se ogni studente distrugge il foglietto che ha staccato dal blocchetto, nessuno può risalire all'età individuale di ognuno di loro, a meno che due studenti del gruppo non decidano di collaborare.

Variazioni ed estensioni

Il sistema può essere adattato per consentire votazioni segrete facendo sommare al risultato uno se il voto è sì e zero se è no. Ovviamente se non vengono rispettate le regole e qualcuno somma più di uno (o un numero negativo) la votazione non è equa. Tuttavia questa frode corre facilmente il rischio di essere rilevata se la somma totale dei voti *sì* diviene maggiore del numero delle persone che hanno votato.

Cosa c’entra tutto questo?

I computer memorizzano tante informazioni personali: il saldo del nostro conto corrente, le nostre reti sociali, quanto dobbiamo pagare di tasse, da quanto tempo abbiamo la patente di guida, la nostra storia creditizia, i risultati degli esami, la nostra cartella clinica, ecc. La privacy è molto importante! Ma dobbiamo al tempo stesso avere la possibilità di condividere alcune di queste informazioni con gli altri. Per esempio quando paghiamo col Bancomat il conto alla cassa consentiamo al gestore del negozio di verificare che c’è denaro a sufficienza nel nostro conto corrente per pagare l’importo richiesto.

Spesso finiamo con fornire più informazioni del necessario. Per esempio, se facciamo una transazione elettronica in un negozio riveliamo anche quale sia la nostra banca, il nostro nome e se paghiamo con un assegno anche il numero del conto corrente. Inoltre la banca scopre in quale negozio abbiamo deciso di fare shopping. Le banche possono creare un profilo del cliente analizzando ciò che acquista, quanto il cliente spende in media e quali sono i luoghi che normalmente visita. Se avessimo pagato con il denaro contante nessuna di queste informazioni sarebbe stata rivelata. Molte persone non si preoccupano troppo delle informazioni che possono essere carpite in questo modo, ma c’è la possibilità che queste informazioni vengano usate per compiere abusi o per spedire pubblicità selettiva (per esempio inviando le offerte delle agenzie di viaggio a chi spende tanto in biglietti aerei) o per discriminare (per esempio fornendo servizi migliori a chi ha la carta riservata ai clienti più ricchi della banca) oppure anche per ricattare (minacciando di rivelare i dettagli di pagamenti che possono creare imbarazzo). In ogni modo, le persone potrebbero anche semplicemente cambiare le loro abitudini di acquisto se pensassero che c’è chi le sta spiando.

Questa perdita di privacy è comunemente accettata dalla maggior parte della popolazione, anche se esistono protocolli crittografici che ci consentirebbero di fare transazioni finanziarie con lo stesso livello di privacy del denaro contante. Potrebbe sembrare difficile da credere che il denaro possa essere spostato dal vostro conto corrente al conto del negozio senza che nessuno sappia nè da dove il denaro venga nè dove stia andando. Questa attività mostra che è plausibile che esista una soluzione a questo problema: entrambe le situazioni richiedono una condivisione limitata di informazioni e nell’esempio precedente abbiamo visto che ciò è possibile a patto di scegliere un protocollo appropriato.

Per ulteriori approfondimenti

Un articolo famoso che mette in evidenza gli argomenti qui trattati è stato scritto da David Chaum, con il titolo provocatorio “Security without

identification: transaction systems to make Big Brother obsolete." (sicurezza senza identificazione: sistemi di transazioni e che rendono obsoleto il grande fratello) [2]. L'articolo, di lettura non eccessivamente impegnativa, fornisce esempi di semplici di protocolli che nascondono le informazioni. Mostra anche come transazioni totalmente private possano essere fatte usando "denaro elettronico". L'articolo compare nel numero di ottobre 1985 della rivista *Communications of the ACM*.

Attività 18

Testa o croce in Perù — *I protocolli crittografici*

Sommario

Questa attività mostra come si possa compiere un'azione semplice ma che sembra impossibile: fare una scelta casuale, come in una partita a testa o croce, fra persone collegate da una linea telefonica e che non necessariamente si fidino l'una dell'altra.

Conoscenze richieste

✓ Logica booleana

✓ Funzioni

✓ Enigmistica

Età

✓ A partire dai 9 anni

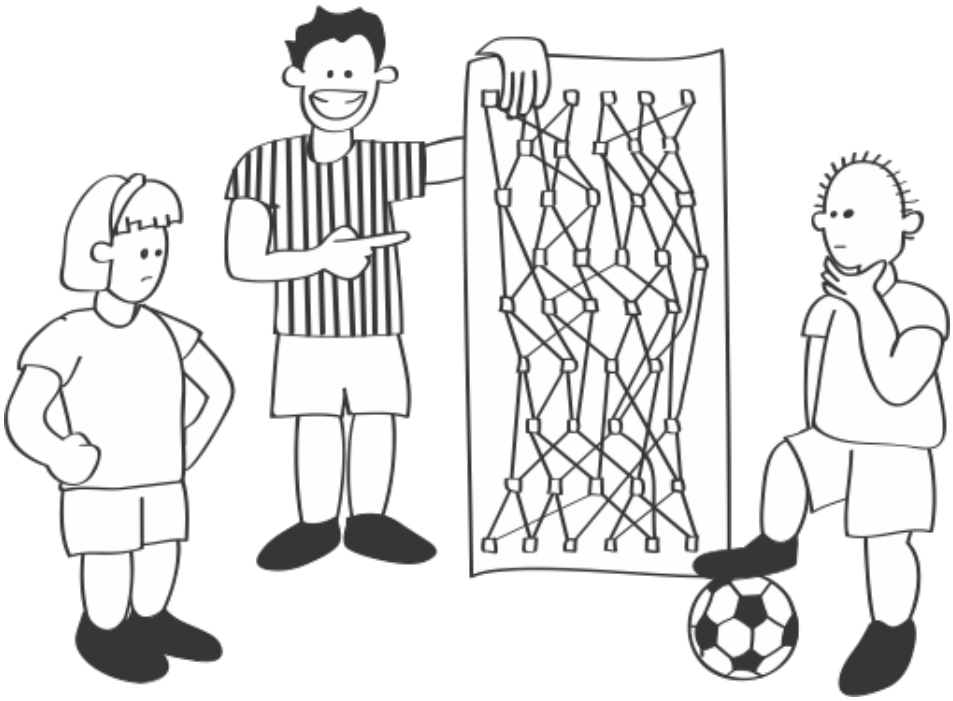
Materiale

Ogni gruppo di studenti deve avere:

✓ una copia del foglio di lavoro: Testa o croce in Perù;

✓ due dozzine circa di piccoli bottoni o gettoni di due diversi colori.

Testa o croce in Perù



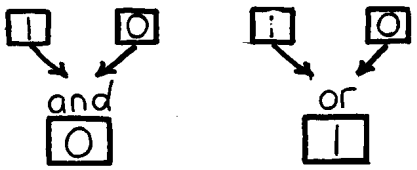
Introduzione

Questa attività è stata inizialmente ideata quando uno degli autori (MRF) stava lavorando in Perù. Questo è il motivo del titolo. È possibile adattare la storia usando nomi e situazioni locali.

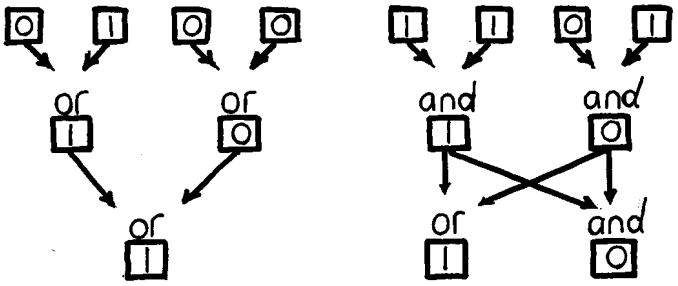
La squadre di calcio di Lima e Cuzco devono decidere chi giocherà in casa per la prossima partita di campionato. Il modo più semplice è di tirare una moneta e giocarsi la scelta a testa o croce. Ma le due città sono distanti e nè Alicia, che rappresenta il Lima, nè Basilio che rappresenta il Cuzco, vogliono spendere il tempo e il denaro per incontrarsi solo per tirare la moneta. Possono prendere la decisione parlando al telefono? Alicia potrebbe tirare la moneta e Basilio decidere se vuole testa o croce. Ma non funzionerebbe perché se Basilio decide "testa" Alicia può semplicemente dire "mi dispiace è venuto croce" senza che Basilio possa controllare. Alicia non è normalmente disonesta ma la partita è importante e le tentazioni sono forti. Anche posto che agisca in modo corretto, se Basilio perde sarà sicuro dell'onestà di Alicia?

Gli studenti saranno in grado di capire meglio questa unità se hanno già imparato la rappresentazione tramite numeri binari (attività 1, conta i punti), il concetto di parità (attività 4, la magia delle carte girate) e se hanno visto esempi di funzioni a senso unico nell'attività 15, la città turistica.

Ecco come Alicia e Basilio decidono di fare. Lavorando insieme, disegnano un circuito fatto di porte-`and` e porte-`or`, come spiegato in seguito. In linea di principio tutto questo può essere fatto al telefono anche se alla fine risulterebbe non poco noioso (si può pensare di usare l'e-mail o il fax). Durante la costruzione entrambi hanno l'interesse a fare in modo che il circuito sia sufficientemente complesso in modo che l'altro non possa barare. Il circuito finale viene pubblicato in modo che tutti lo possano vedere.

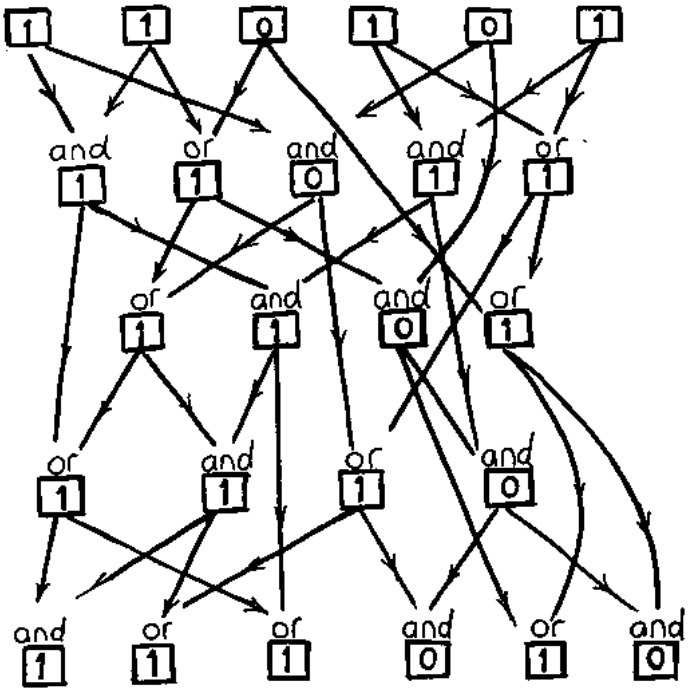


Le regole delle porte `and` e `or` sono semplici. Ogni porta ha due entrate (input) e un'uscita (output). Ogni segnale fornito in ingresso assume uno dei due valori, 0 oppure 1, che possono essere interpretati rispettivamente come falso (false) e vero (true). L'uscita di una porta `and` vale 1 (vero/true) solo quando entrambe le entrate valgono 1 (vero/true). Di conseguenza l'uscita varrà 0 (falso/false) in tutti gli altri casi. L'output di una porta `or` vale 0 (falso/false) solo se entrambi i dati in ingresso valgono 0 (falso/false); se uno o entrambi gli ingressi valgono 1 (vero/true) allora il risultato sarà 1 (vero/true).



L'uscita di una porta può essere collegata all'ingresso di una porta (o a più porte) per produrre un effetto più complesso. Per esempio nella figura alla sinistra l'uscita delle due porte `or` diventano dati in ingresso per una terza porta or. Nel circuito alla destra le uscite di entrambe le porte `and` in alto alimentano l'input delle due porte in basso: l'intero circuito ha quindi quattro ingressi e due uscite.

Per il gioco del "testa o croce in Perù" abbiamo necessità di un circuito ancora più complesso. Il circuito disegnato nel foglio di lavoro ha sei ingressi e sei uscite. Riportiamo qui di seguito come esempio i calcoli relativi ad una specifica configurazione dei valori in ingresso.



Per giocare a testa o croce al telefono tramite questo circuito si fa così: Alicia sceglie a caso un numero di sei bit, lo annota e lo tiene segreto. Mette i sei bit come valori di ingresso nel circuito e spedisce a Basilio i sei valori di uscita. Quando Basilio riceve questi valori tenta di indovinare la parità del numero segreto di Alicia. Se il circuito è sufficientemente complesso Basilio non potrà indovinare quale sia il numero segreto di Alicia e quindi la scelta dovrà essere casuale (potrebbe giocare da solo a testa o croce per determinare la scelta). Basilio vince e quindi la partita si svolgerà a Cuzco se la sua risposta si rivelerà corretta. Alicia vince se la scelta di Basilio si rivelerà errata. Quando Basilio ha detto ad Alicia la sua scelta, Alicia può rivelare il suo numero segreto così che Basilio possa controllare e confermare la correttezza del risultato ottenuto.

1. Dividete gli studenti in piccoli gruppi, date ad ogni gruppo il circuito e alcuni gettoni e spiegate loro la storia. La situazione sarà probabilmente più convincente se gli studenti immagineranno i capitani delle loro squadre giocare a testa o croce coi capitani delle squadre delle scuole rivali. Stabilite una convenzione per i colori dei gettoni (per esempio il rosso è 0 e il blu è 1) e fate contrassegnare o colorare la legenda in alto.

2. Mostrate agli studenti come mettere i gettoni sugli input in modo da mostrare i numeri che Alicia sceglie. Quindi spiegate le regole delle porte `and` e `or` che sono riportate in fondo al foglio (valutate l'idea di far colorare i valori dei bit in modo coerente col colore dei gettoni).

3. Mostrate come funziona il circuito, mettendo gettoni nei nodi in

modo da poter derivare le uscite delle porte corrispondenti. Questa operazione deve essere fatta con accuratezza; la tavola seguente (che non deve essere data agli studenti) mostra gli output per ogni possibile configurazione di input. Serve per vostra referenza, da consultare in caso di dubbio.

Input	000000	000001	000010	000011	000100	000101	000110	000111
Output	000000	010010	000000	010010	010010	010010	010010	010010
Input	001000	001001	001010	001011	001100	001101	001110	001111
Output	001010	011010	001010	011010	011010	011010	011010	011111
Input	010000	010001	010010	010011	010100	010101	010110	010111
Output	001000	011010	001010	011010	011010	011010	011010	011111
Input	011000	011001	011010	011011	011100	011101	011110	011111
Output	001010	011010	001010	011010	011010	011010	011010	011111
Input	100000	100001	100010	100011	100100	100101	100110	100111
Output	000000	010010	011000	011010	010010	010010	011010	011010
Input	101000	101001	101010	101011	101100	101101	101110	101111
Output	001010	011010	011010	011010	011010	011010	011010	011111
Input	110000	110001	110010	110011	110100	110101	110110	110111
Output	001000	011010	011010	011010	011010	111010	011010	111111
Input	111000	111001	111010	111011	111100	111101	111110	111111
Output	001010	011010	011010	011010	011010	111010	011010	111111

4. Ora ogni gruppo deve eleggere una Alicia e un Basilio. Alternativamente, il gruppo si può spezzare in due parti e ogni sottogruppo fare le veci di Alicia o di Basilio. Alicia deve scegliere un numero random da mettere in ingresso nel circuito, deve quindi calcolare i valori di uscita (output) e comunicarli a Basilio. Basilio deve indovinare la parità del numero usato da Alicia come ingresso del circuito (cioè se ha un numero pari o dispari di valori posti a uno/vero/true). Sarà chiaro che la decisione di Basilio sarà casuale. Alicia a questo punto può mostrare il numero mantenuto segreto e Basilio vincerà se avrà indovinato la parità corretta. A questo punto Basilio può controllare che Alicia non abbia modificato il numero: sarà sufficiente controllare con il circuito che fornisca lo stesso output che Alicia aveva comunicato.

   A questo punto la partita di “testa o croce” è terminata.

   Basilio potrebbe barare se riuscisse a indovinare il numero di sei bit usato da Alicia per generare l’output ricevuto. L’interesse di Alicia è quindi di usare una funzione a senso unico (come discusso nell’attività 15) per evitare che Basilio possa barare. Una funzione a senso univo (one-way) è facile da calcolare ma è difficile sapere quale valore di ingresso abbia determinato uno specifico valore di uscita.

   Alicia potrebbe barare se riuscisse a scoprire due valori di diversa

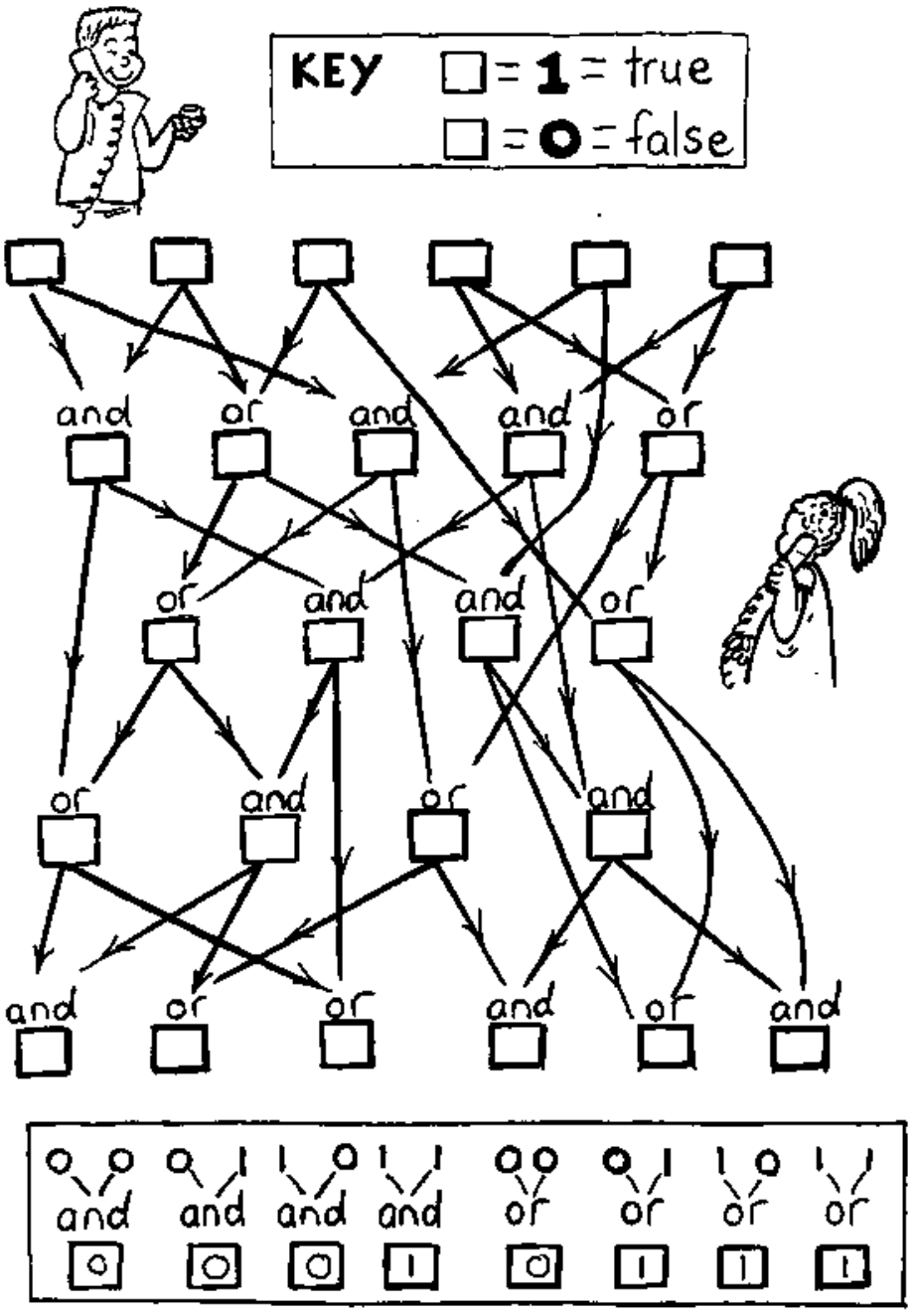
parità che producono esattamente gli stessi valori in uscita. Così facendo, qualsiasi fosse la scelta di Basilio, Alicia potrebbe sostenere che ha sbagliato ad indovinare la parità. L'interesse di Basilio è di assicurare che il circuito non faccia corrispondere configurazioni di ingresso differenti agli stessi valori in uscita o almeno che sia difficile trovare queste corrispondenze.

5. Provate a vedere se gli studenti riescono a trovare casi nei quali Alicia o Basilio possano barare. Per esempio nella prima riga della tabella si vede che molteplici configurazioni di input generano in output 010010, per esempio 000001, 000011, 000101, etc. Quindi se Alicia dichiara a Basilio il numero in output 010010 può poi scegliere 000001 se Basilio dice "pari" e 000011 se dice "dispari".

   Con questo circuito è difficile barare per Basilio. Ma se per esempio l'output è 011000, allora l'input di Alicia non può che essere 100010, non c'è nessun'altra possibilità (potete controllare nella tabella). Quindi se questo è il numero che Alicia ha dato a Basilio, quest'ultimo può scegliere la parità pari ed è sicuro di vincere. Nella realtà se si usasse un computer per risolvere questo problema i numeri in input e in output avrebbero un numero di bit molto maggiore e quindi ci sarebbero troppe configurazioni da provare (a ogni bit aggiunto raddoppia il numero delle configurazioni) per trovare un caso simile a questo.

6. Ora chiedete ai gruppi di studenti di ideare i loro circuiti per questo gioco. Guardate se riescono a creare circuiti per i quali sia facile barare per Alicia e altri nei quali sia facile barare per Basilio. Nulla vieta di avere un numero di bit in ingresso e in uscita diversi fra loro e diversi da sei.

Foglio di lavoro: testa o croce in Perù

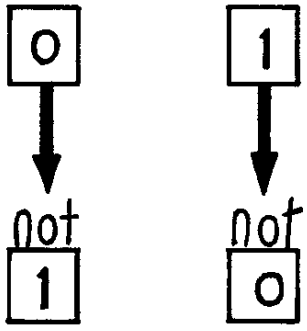


Variazioni ed estenzioni

1. Un chiaro problema pratico è la cooperazione necessaria per costruire un circuito che sia accettabile sia per Alicia sia per Basilio. Questa può essere una fase divertente per gli studenti, ma può rendere la procedura inapplicabile in pratica, in modo particolare se gli accordi devono essere presi per telefono. In ogni caso c'è una semplice alternativa: sia Alicia sia Basilio costruiscono un circuito ed entrambi rendono pubblico lo schema del proprio circuito. Alicia fornisce il proprio input ad entrambi i circuiti e combina le due sequenze di output in questo modo: il valore finale sarà uno se i bit corrispondenti hanno lo stesso valore, zero in caso contrario. In questo modo nessuno dei due partecipanti può barare (a meno che non barino entrambi) se anche una delle due funzione non fosse a senso unico, la combinazione delle due lo sarebbe.

Le prossime due variazioni non sono relative ai protocolli crittografici o al problema di giocare a testa o croce, bensì all'idea del circuito costruito mediante le porte logiche `and` e or. Verranno esplorate alcune importanti nozioni non solo relative ai circuiti per l'elaborazione ma alla logica stessa. Questo tipo di logica viene chiamata *Algebra di Boole* dal nome del matematico George Boole (1815-64).

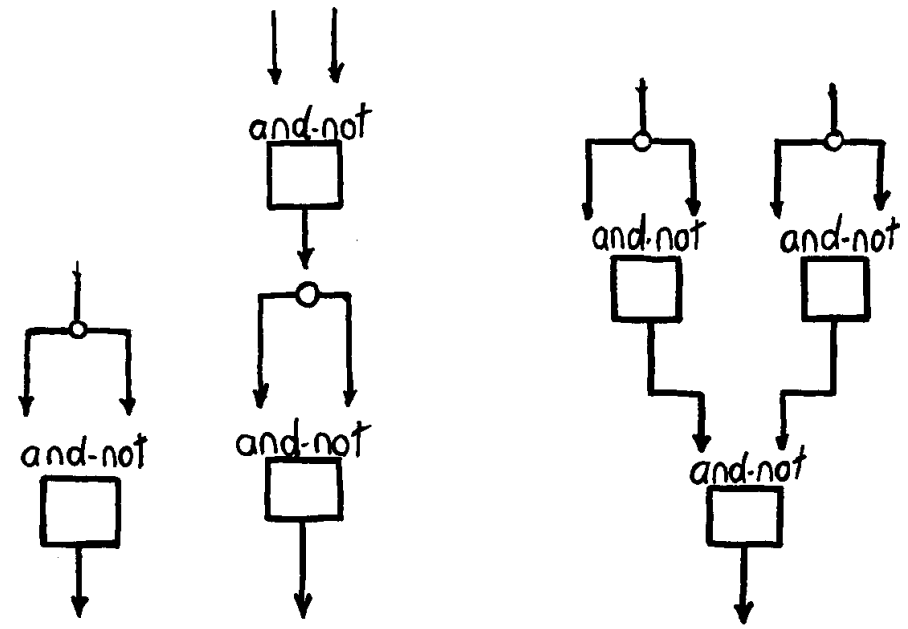
2. Gli studenti potranno aver notato che la configurazione formata solo da input di valore zero, 000000, produce in output sempre 000000 e, in maniera duale, la configurazione con tutti gli input uguali ad uno, 111111, produce in output 111111 (ci possono essere altre configurazioni di input che forniscono questi output, per esempio 000010 produce 000000 mentre 110111 produce 111111). Questa proprietà deriva dal fatto che il nostro circuito usa solamente porte `and` e `or`. Se si aggiunge un nodo `not` che prende in input un solo bit e ne inverte il valore in uscita ($0 \to 1$, $1 \to 0$), gli studenti possono costruire circuiti che non hanno questa proprietà.



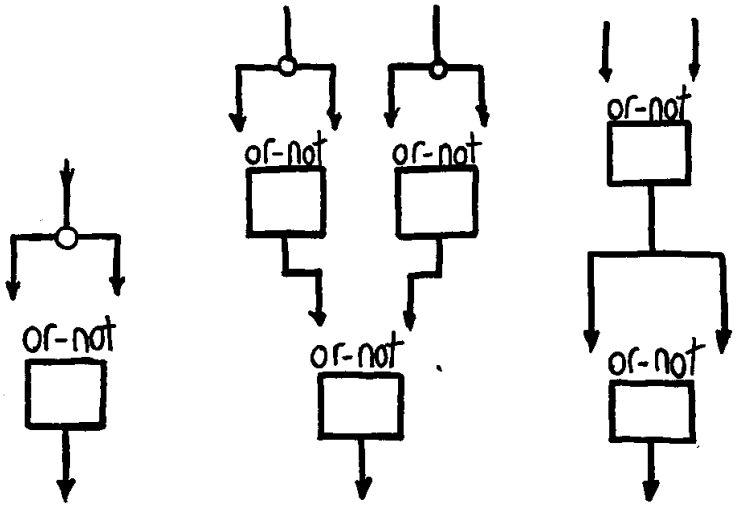
3. Altri due circuiti importanti sono gli `and-not` e gli `or-not` (normalmente abbreviati rispettivamente coi nomi di `nand` e `nor`) che sono come `and` e or ma sono seguiti da un `not`. Quindi a nand $b =$ not$(a$ and $b)$. L'introduzione di questi circuiti non permette di costruire nuovi tipi di circuito, più di quelli realizzabili con le porte che abbiamo già studiato, perché il loro effetto potrebbe essere sempre ricostruito usando porte `and`, `or` e `not`. In ogni caso, questi circuiti hanno una importante proprietà: usando solamente porte di tipo `nand` (o similmente solo porte di tipo `nor`) è possibile realizzare tutti gli altri circuiti.

Avendo introdotto le porte `nand` e `nor`, una sfida per gli studenti è quella di trovare il modo di realizzare tutte le porte mediante altre e arrivare a realizzare ogni tipo di porta connettendo più porte dello stesso tipo.

Le porte `not`, `and` e `or` possono essere costruite a partire dalle porte `nand` (in alto) o con sole porte `nor` (in basso).



(a) (b) (c)



(d) (e) (f)

Cosa c'entra tutto questo?

Negli ultimi anni abbiamo assistito ad un enorme aumento del commercio elettronico ed è essenziale garantire uno scambio sicuro del denaro, delle transazioni riservate nonché la gestione di documenti firmati e con valore legale. Lo scopo della crittografia è di consentire una comunicazione sicura e riservata. Alcuni decenni fa i ricercatori scoprirono un risultato controintuitivo: la segretezza può essere garantita anche se certe informazioni sono pubbliche. Il risultato è la "crittografia a chiave pubblica" descritta nell'attività 19, Kid Krypto, che è oggi ampiamente usata come strumento principale per lo scambio sicuro di informazioni. Per esempio potete aver impostazioni nel vostro browser web per attivare SSL (Secure Sockets Layer) o anche TLS (Transport Layer Security); questi sistemi sono basati sulla crittografia a chiave pubblica e abilitano il vostro browser a creare connessioni sicure con certi siti web quali la vostra banca. La comunicazione in questi casi sarebbe sicura anche se ci fosse qualcuno che sta intercettando il vostro traffico sulla rete.

La crittografia non serve solo per mantenere le cose segrete, ma anche per porre limiti a ciò che gli altri possono scoprire e anche per creare "affidabilità" fra persone che sono geograficamente distanti. Le regole formali (dette "protocolli") per le transazioni crittografiche sono state ideate per permettere questi apparentemente impossibili servizi, come le firme digitali non falsificabili o l'abilità di possedere un segreto (come una password) e farne uso senza rivelare quale esso sia. Giocare a testa o croce al telefono è un problema analogo sebbene più semplice. Anch'esso sembra essere impossibile, ma come abbiamo visto non è così.

Nelle situazioni reali Alicia e Basilio non disegnerebbero un circuito per conto proprio ma userebbero un programma capace di costruire "logicamente" il circuito. Probabilmente nessuno dei due sarebbe interessato ai dettagli costruttivi del programma, ma entrambi vorrebbero essere sicuri che l'altro non possa influenzare l'esito della decisione indipendentemente da quanto sia esperto con l'informatica o da quanto tenacemente ci provi.

In linea di principio, ogni disputa potrebbe essere risolta facendo affidamento ad un giudice terzo imparziale. Il giudice potrebbe ricevere il circuito, il numero random usato da Alicia, il messaggio mandato a Basilio con l'output del circuito e la scelta fatta da Basilio rispedita come risposta ad Alicia. Alla fine della transazione tutte queste sono informazioni pubbliche quindi tutti i partecipanti dovranno concordare sui dati che hanno contribuito al risultato. Il giudice potrà immettere il numero di Alicia nel circuito e controllare che l'output sia quello spedito a Basilio e quindi stabilire se la decisione sia stata presa in modo

corretto o meno. È forse superfluo dire che il fatto importante è che esiste una chiara procedura di verifica della corretta applicazione delle regole ed è quindi molto improbabile che una disputa possa mai nascere. Se invece Alicia e Basilio avessero giocato a testa o croce con una reale moneta, nessun giudice sarebbe potuto intervenire in caso di disputa.

Un circuito con pochi componenti come quello illustrato in questa attività non potrebbe essere applicato a casi reali perché è possibile creare una tabella che comprenda tutti i casi possibili e usarla per barare. Usando trentadue bit come imput si potrebbe avere una maggior protezione. Questo però non potrebbe garantire che sia difficile barare, dipende da come è fatto il circuito. Altri metodi possono essere basati sulla funzione a senso unico definita nell'attività 15, la città turistica. I metodi usati in pratica sono basati sulla "fattorizzazione di numeri molto grandi", che è nota come un problema molto complesso (anche se, come vedremo alla fine della prossima attività, non è *NP-completo*). È facile controllare se un numero è un fattore di un altro, ma trovare ii fattori di un numero è un'attività che richiede tantissimo tempo. Questo rende molto complesso per Alicia e Basilio (e per il giudice) poter fare i calcoli a mano quindi, come notato sopra, tutti i calcoli vengono svolti da programmi di uso comune.

Le firme digitali sono basate su una simile idea. Alicia, rendendo pubblico l'output del circuito per uno specifico input segreto che lei ha scelto, può provare in un secondo tempo che è stata lei a generare quell'output. Con una reale funzione a senso unico nessun altro può produrre un input capace di generare lo stesso messaggio. Nessun impostore può farsi passare per Alicia. Per fare una reale firma digitale è necessario un protocollo più complesso per garantire che Alicia possa firmare un messaggio e anche per garantire che gli altri possano controllare l'autenticità della firma di Alicia anche nel caso che Alicia stessa sostenga di non averlo firmato. Il principio è in ogni caso lo stesso.

Un'altra applicazione consente di giocare a poker al telefono, senza alcun arbitro che dia le carte e registrando tutte le *mani* di entrambi giocatori. Tutto deve essere fatto dai giocatori stessi, con il ricorso ad un giudice alla fine del gioco solo in caso di disputa. Situazioni simili accadono seriamente nelle contrattazioni commerciali. Ovviamente i giocatori devono tenere le loro carte segrete durante il gioco. Ma devono anche essere onesti, non possono sostenere di avere un asso se non l'hanno veramente. Questo può essere controllato attendendo la

fine del gioco e consentendo poi ad ogni giocatore di guardare la prima mano di tutti gli altri e la sequenza di azioni fatte. Un altro problema è come fare a tenere tutte le carte di ogni giocatore segrete fino alla fine del gioco. Sorprendentemente è possibile consentire questo usando un protocollo crittografico non molto diverso dal gioco a testa o croce qui descritto.

I protocolli crittografici sono estremamente importanti nelle transazioni elettroniche, per identificare il titolare di una carta di credito, per autorizzare l'uso di un telefono cellulare, o per autenticare il mittente di un messaggio di e-mail. L'abilità di fare queste azioni in modo affidabile è cruciale per il successo del commercio elettronico.

Per ulteriori approfondimenti

Il libro di Harel con titolo *Algorithmics* [5] discute le firme digitali e i protocolli crittografici correlati. Mostra anche come giocare a poker al telefono, una idea introdotta per la prima volta nel 1981 in un capitolo dal titolo "mental poker" del libro *The Mathematical Gardener* curato a D.A. Klarner [8]. *Cryptography and data security* di Dorothy Denning [11] è un testo scientifico eccellente sulla crittografia. *Turing Omnibus* di Dewdney [3] ha una sezione dedicata alla logica di Boole che discute i blocchi usati per la costruzione del circuito presentato in questa attività.

Attività 19

Kid Krypto — *La crittografia a chiave pubblica*

Sommario

La crittografia è il concetto chiave della sicurezza delle informazioni. E il concetto fondamentale della crittografia moderna è che usando solo informazioni pubbliche un mittente può chiudere a chiave il messaggio in uno scrigno (logico) che solo il legittimo destinatario potrà (privatamente) aprire. È come se ogni studente comprasse un lucchetto, ci scrivesse sopra il proprio nome e lo lasciasse aperto sul tavolo tenendo la chiave. I lucchetti ai quali facciamo riferimento sono del tipo che si usa normalmente per la catena della bicicletta o per chiudere l'armadietto della piscina, non hanno necessità della chiave per venir chiusi, basta stringere il ferro a *U* fino a che non fa click. Se ora voglio spedire un messaggio a uno studente, cerco sul tavolo il lucchetto con il suo nome e uso il lucchetto per chiudere una scatola nella quale avrò messo il messaggio. Anche se la scatola cadesse nelle mani sbagliate nessuno potrà aprire la scatole se non il destinatario legittimo che ha la chiave del lucchetto. Con questo schema non è necessaria alcuna precedente forma di comunicazione, non servono chiavi segrete.

Questa attività mostra come questo può essere fatto in modo digitale. E nel mondo digitale non è necessario prendere il lucchetto dal tavolo, i lucchetti logici possono essere copiati. Se dovessimo costruire una copia di un lucchetto fisico, dovremmo smontarlo e inevitabilmente scopriremmo come funziona a magari potremmo forgiare una chiave falsa. Ma nel mondo digitale è possibile consentire alle persone di copiare i lucchetti senza che possano scoprire la chiave! Sembra impossibile? Continuate a leggere.

Abilità

✓ Saper risolvere giochi enigmistici

Età

✓ A partire da 11 anni.

Materiale

Create gruppi di circa quattro studenti. In ogni gruppo individuate due sottogruppi. A ogni sottogruppo viene data una copia del foglio di lavoro *Mappe Kid Krypto*. Quindi ogni gruppo ha necessità di

✓ due copie delle Mappe Kid Kripto (pag. 235)

Servirà anche:

- ✓ uno strumento per mostrare alla classe lo schema Codifica Kid Krypto (pag. 236)
- ✓ uno strumento per disegnare note sullo stesso schema.

Kid Krypto

Introduzione

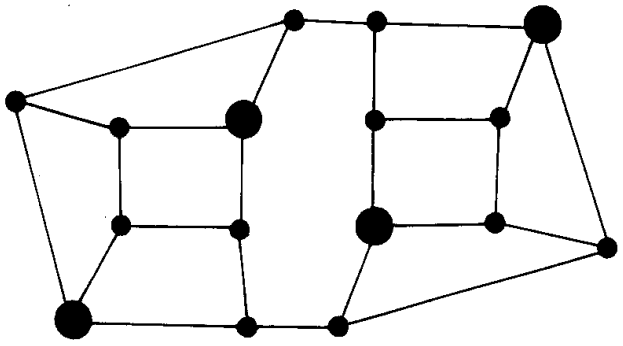
Questa è l'attività più impegnativa dell'intero libro dal punto di vista tecnico. Anche se può essere gratificante, richiede tanta attenzione e tanta concentrazione per avere successo. Gli studenti dovrebbero aver già studiato gli esempi di funzioni a senso unico dell'attività 15, la città turistica ed è di aiuto se hanno già completato le altre attività di questa sezione (l'attività 17, condividere i segreti e l'attività 18, la monetina peruviana. Questa atività usa anche idee introdotte nelle attività 1, conta i punti e 5, indovina indovinello).

Anna deve mandare a Bruno un messaggio segreto. Normalmente noi pensiamo che il messaggio segreto sia una frase o un breve testo. Nell'esercizio seguente Anna spedirà un carattere o meglio un numero che rappresenta un carattere. Anche se questo messaggio può sembrare semplicistico, tenete a mente che Anna può spedire un'intera sequenza di questi "messaggi" per formare una frase e che questo lavoro verrà fatto in realtà da un computer. Vedremo come nascondere il numero di Anna in un messaggio criptato con la chiave di Bruno in modo tale che se anche qualcuno intercettasse il messaggio non sarebbe in grado di decodificarlo. Solo Bruno, che ha la chiave del lucchetto, potrà leggerlo.

Useremo delle mappe per nascondere il messaggio. Non sono le mappe del tesoro, dove una X indica il punto esatto dove scavare, ma piuttosto mappe stradali come quelle della città turistica 15, dove le linee sono le strade e i punti gli incroci. Ogni mappa ha una versione pubblica, cioè il lucchetto, e una versione privata, cioè la chiave.

Discussione

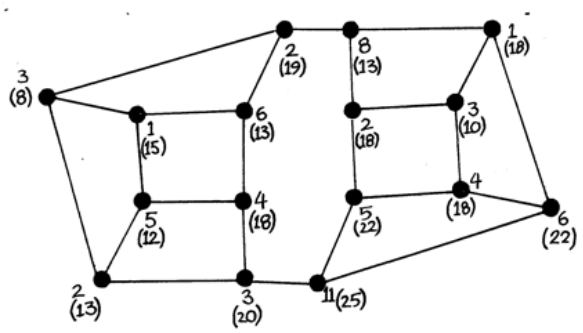
Nel foglio di lavoro Codifica Kid Krypto (pag. 236) c'è la chiave pubblica di Bruno. Non è un segreto, Bruno può mettere questa mappa sul tavolo (o su una pagina web) perché tutti possano vederla oppure (equivalentemente) può darne una copia a chiunque voglia spedirgli un messaggio. Anna ha una copia della mappa pubblica di Bruno così come tutti gli altri. La figura qui a destra mostra la chiave privata di Bruno. È uguale alla sua chiave pubblica eccetto che per il fatto che alcuni degli incroci sono stati



evidenziati disegnando un punto di dimensione maggiore. Bruno deve mantenere segreta questa mappa.

Questa attività è bene farla tutti insieme in classe, almeno all'inizio, perché richiede una notevole quantità di lavoro. Anche se le operazioni non sono particolarmente difficili occorre svolgerle con grande accuratezza. Ogni errore può far fallire l'esperimento. È importante che gli studenti si rendano conto di come sia sorprendente che questo tipo di crittografia possa esistere, sembra infatti impossibile perché questo stupore fornisce la forza di arrivare al risultato nonostante lo sforzo richiesto. Un elemento che fornisce molta motivazione agli studenti è che questo metodo può consentire loro di scambiare messaggi segreti all'interno della classe e anche se l'insegnante sa come il messaggio è codificato, non sarà in grado di decodificarlo.

1. Mostrate la mappa pubblica di Bruno (il foglio di lavoro *codifica kid crypto* a pag. 236). Decidete quale numero Anna voglia inviare. A questo punto scrivete numeri casuali in ogni incrocio nella mappa in modo che la somma di tutti i numeri sia il numero da trasmettere (sono i numeri indicati senza parentesi nella figura). Anna nell'esempio ha scelto di inviare 66, quindi la somma di tutti i numeri senza parentesi è 66. Potete usare anche numeri negativi, se volete, per ottenere il valore cercato.



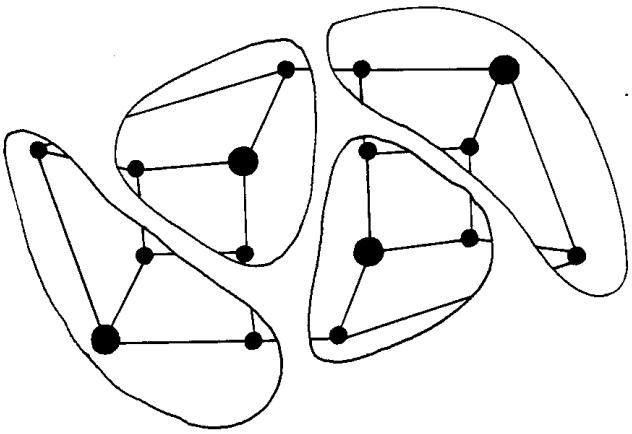
2. Ora Anna deve calcolare i valori da mandare a Bruno. Se spedisse i numeri scritti sulla mappa chiunque leggendo il messaggio potrebbe sommare i numeri assieme ed ottenere 66, il messaggio segreto. Non saranno questi i numeri inviati a Bruno.

   Invece, per ogni incrocio considerate l'incrocio e tutti gli incroci "vicini", quelli che si raggiungono percorrendo una sola strada, e sommate i valori ottenuti (quelli cioè nell'incrocio stesso e in tutti i vicini). Scrivete questi numeri fra parentesi, come in figura, o usando una penna di diverso colore. Per esempio se esaminiamo l'incrocio (cioè il nodo) più a destra nella figura vediamo che ha il valore 6 e ha tre incroci vicini che hanno valore 1, 4 e 11. Il totale è quindi 22. Ora ripetete questo procedimento per tutti gli incroci della mappa: dovreste ottenere tutti i numeri indicati fra parentesi nella figura.

3. Anna spedirà a Bruno la mappa con solamente i numeri fra parentesi. Cancellate quindi i numeri senza parentesi (e le note dei

calcoli, se ci sono!) lasciando solo i numeri da inviare (o prendete una nuova mappa e ricopiate solo questi numeri). Provate a vedere se qualche studente è in grado di dire quale fosse il numero originale del messaggio. Nessuno sarà in grado.

4. Solo chi è in possesso della chiave privata di Bruno può decodificare il messaggio e trovare il numero che Anna ha inviato. Per decodificare il messaggio è sufficiente che Bruno prenda la mappa segreta e sommi fra loro i numeri presenti nei punti evidenziati (quelli più grandi). Nell'esempio questi incroci contengono i valori 13, 13, 22 e 18 che sommati insieme danno proprio 66, il numero spedito da Anna.



5. Ma come funziona? La mappa è speciale. Supponete che Bruno prenda la sua mappa e tracci una linea attorno ad ogni incrocio evidenziato comprendente anche gli incroci "vicini". Provate a disegnare anche voi queste linee come nella figura a lato. Queste linee partizioneranno i nodi in modo che tutti gli incroci apparterranno a un pezzo (ed a uno solo, mai a due). Quindi sommare insieme i numeri di questi quattro incroci equivale a sommare tutti i numeri senza parentesi della prima mappa di Anna e quindi il risultato sarà il messaggio segreto.

Uffa! È stato necessario un sacco di lavoro per spedire una sola lettera. E se è necessario così tanto lavoro per una lettera la crittografia è veramente faticosa. Ma guardate ciò che avete ottenuto: completa segretezza usando una chiave pubblica, senza che i partecipanti abbiano dovuto mettersi d'accordo in anticipo per una chiave segreta. Potete pubblicare la vostra chiave pubblica in una bacheca e chiunque può usarla per spedirvi messaggi segreti, che nessuno è in grado di leggere se non conosce la chiave segreta. E nella vita reale tutti i calcoli vengono svolti da un programma, molto spesso da un modulo del vostro programma browser per il web, quindi è solo il vostro computer che deve fare il duro lavoro di calcolo.

Forse la vostra classe sarà felice di sapere che siete riusciti ad entrare nel club esclusivo di quanti hanno effettivamente provato un algoritmo di crittografia facendo i calcoli a mano. Normalmente gli studiosi lo considerano un compito quasi impossibile e poche persone lo hanno fatto.

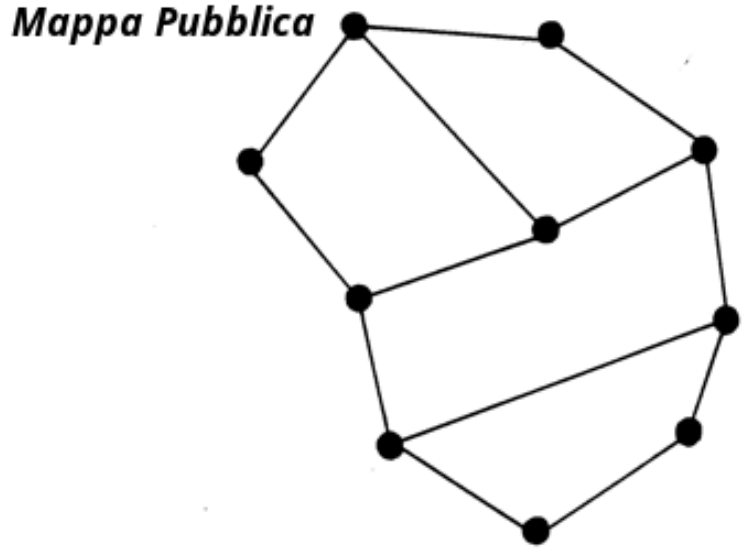
E cosa dire delle intercettazioni? La mappa usata da Bruno è simile a quella dell'attività della città turistica (attività 15), dove gli incroci evidenziati sono quelli che consentono di avere il furgoncino dei gelati ad un isolato di distanza al massimo. Se è facile per Bruno calcolare una nuova mappa partendo dai pezzi indicati nella mappa privata, è molto difficile per gli altri trovare il modo ottimale di porre i furgoncini a partire dalla mappa pubblica, a meno di non usare il metodo di *forza bruta* provando cioè ogni possibile scelta, prima provando con un furgoncino, poi con due e così via fino a che non venga trovata una soluzione. Nessuno sa se esista un modo migliore per una mappa generica, anche se tante persone hanno provato a trovarne uno.

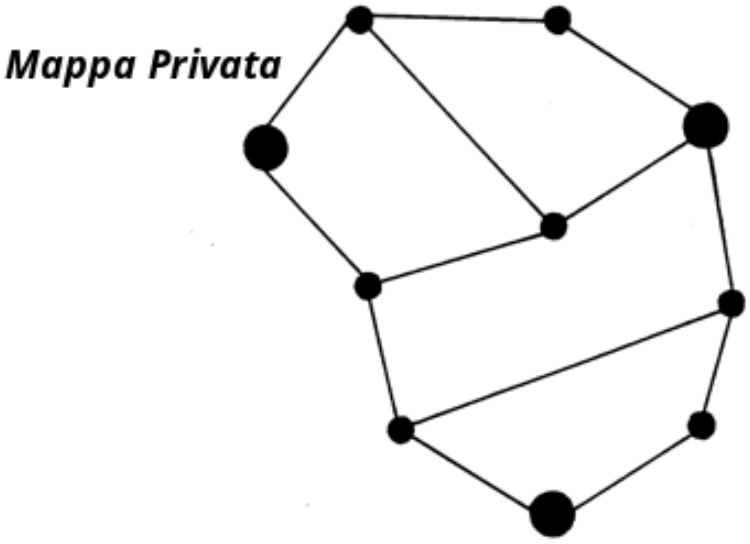
Posto che Bruno usi una mappa sufficientemente complicata con, per esempio, cinquanta o cento incroci, sembra che nessuno possa decodificare il codice. Anche i matematici più esperti hanno tentato e hanno fallito (c'è però una precisazione da fare: vedi oltre nel paragrafo "Cosa c'entra tutto questo?)

7. Ogni coppia dovrebbe scegliere un "messaggio" (un numero intero), codificarlo con la chiave pubblica e dare la mappa risultante all'altro gruppo. L'altro gruppo dovrebbe tentare di decodificare il messaggio, ma non hanno speranza di poter aver successo se non avrete dato loro la mappa segreta. Quindi date loro la mappa segreta e vedete se ora riescono a decodificare correttamente il messaggio.

8. Ora ogni coppia può disegnare la sua mappa, tenendone riservata la versione segreta e dando la versione pubblica all'altra coppia del gruppo o "pubblicandola" sulla lavagna. Il metodo per costruire le mappe è lo stesso usato nell'attività della città turistica. Potete inserire tutte le strade che volete per nascondere la soluzione. State solo attenti a non inserire strade che iniziano o finiscono in uno dei punti "speciali", in questo modo creereste nuovi incroci che porterebbero a raggiungere il furgoncino dei gelati percorrendo una sola strada. Questo nella situazione della città turistica non sarebbe un problema ma *scompiglierebbe* i calcoli della crittografia. Infatti, i punti speciali non partizionerebbero la mappa in parti che non si sovrappongono, che è il punto essenziale che consente al metodo di funzionare.

Foglio di lavoro: le mappe di Kid Krypto

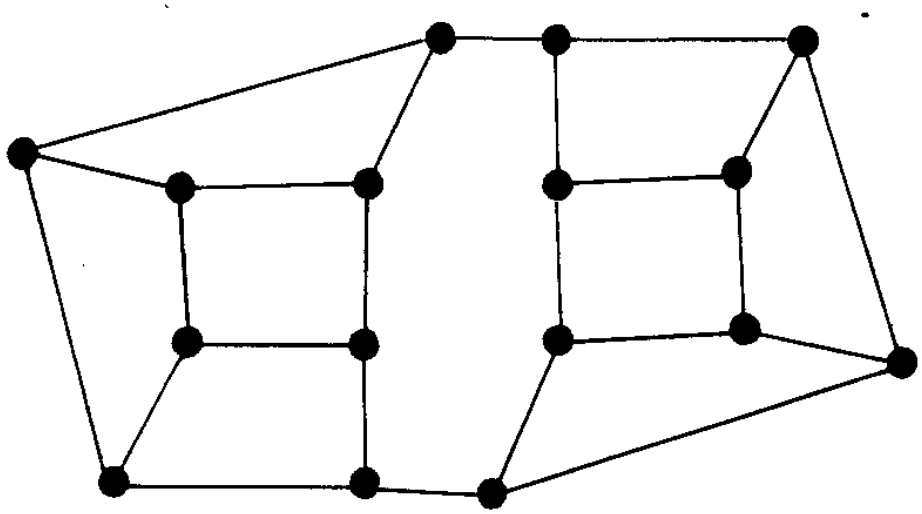




Usate queste mappe per criptare e decriptare i messaggi, come indicato nel testo.

Foglio di lavoro: la codifica Kid Krypto

Mostrate questa “mappa” alla classe e usatela per dimostrare la codifica di un messaggio



Cosa c'entra tutto questo?

È chiaro che tutti noi vogliamo spedire messaggi segreti attraverso le reti di computer e nessuno, se non i legittimi destinatari dei nostri messaggi, deve essere in grado di decodificare i messaggi, non importa quanto gli intercettatori siano furbi o quanto tenacemente ci abbiano provato. Naturalmente esistono molti modi per scambiare messaggi riservati quando il mittente e destinatario condividono un codice segreto. Ma la parte intelligente della crittografia a chiave pubblica è che Anna può mandare a Bruno un messaggio in modo riservato senza aver fatto alcun accordo segreto prima, basta che Anna abbia preso il "lucchetto" di Bruno da un luogo pubblico come per esempio da una pagina web.

La riservatezza è solo un aspetto della crittografia. L'altra faccia della medaglia è l'autenticazione. Quando Anna riceve un messaggio da Bruno, come può essere sicura che quel messaggio provenga veramente da Bruno e non da qualche impostore? Supponete che Anna riceva un messaggio di posta elettronica che dice: *"cara Anna, sono bloccato senza soldi, per cortesia accredita 100 euro sul mio conto corrente numero 0241-45-783239. Grazie, ciao. Bruno"*. Alcuni sistemi di crittografia a chiave pubblica possono essere usati anche per questo. In questi sistemi è possibile usare le chiavi anche in senso inverso: usare la chiave privata per criptare e quella pubblica per decriptare. Quindi, come Anna può mandare a Bruno un messaggio segreto criptato usando la chiave pubblica di Bruno, Bruno può criptare un messaggio con la sua chiave privata e spedirlo ad Anna. Se Anna riuscirà a leggerlo usando la chiave pubblica di Bruno, sarà certa che provenga veramente da Bruno. Ovviamente il messaggio di Bruno potrà essere decriptato da tutti quelli che hanno la sua chiave pubblica e quindi non sarà un messaggio riservato, ma sarà autenticato. Se il messaggio da spedire è riservato nulla vieta di criptarlo nuovamente usando la chiave pubblica di Anna. Con questo doppio livello di criptazione si avrà al tempo stesso la certezza che il messaggio provenga veramente da Bruno e che nessuno se non Anna possa leggerlo. Segretezza ed autenticazione vengono forniti dallo stesso metodo a chiave pubblica e privata.

È arrivato il momento di ammettere che mentre lo schema illustrato in questa attività è molto simile a quello usato nei veri metodi crittografici a chiave pubblica, non è in realtà sicuro anche usando mappe molto grandi.

La ragione è che, anche se nessuno conosce il modo ottimale di posizionare i camioncini dei gelati su una mappa arbitraria e da questo punto di vista lo schema è sicuro, accade che ci sia un modo differente per attaccare questo schema. È improbabile che l'idea venga ai vostri studenti, almeno fino alla scuola secondaria, ma è bene che sappiate

che esiste. Si può dire che il metodo illustrato è a prova di studente ma non è a prova di matematico. Per cortesia evitate di leggere il prossimo paragrafo se non siete particolarmente inclini alla matematica.

Numerate gli incroci della mappa $1, 2, 3, \ldots n$. Denotate i numeri assegnati all'inizio da Anna con le lettere $b_1, b_2, b_3, \ldots b_n$ (quelli che sommati assieme danno il messaggio segreto) e indicate con $t_1, t_2, t_3, \ldots t_n$ i numeri trasmessi (quelli indicati fra parentesi). Ora se per esempio l'incrocio 1 è connesso con gli incroci 2, 3 e 4, il numero trasmesso relativo al nodo 1 sarà:

$$t_1 = b_1 + b_2 + b_3 + b_4$$

Naturalmente simili equazioni esistono per ogni altro incrocio. È quindi possibile scrivere le equazioni per ogni nodo ottenendo così un sistema lineare di n equazioni in n incognite che può essere risolto con un programma specifico. In questo modo dai valori $t_1, t_2, t_3, \ldots t_n$ e dalla struttura del grafo si può risalire ai valori delle incognite $b_1, b_2, b_3, \ldots b_n$ e sommando insieme questi ultimi al messaggio originale. Lo sforzo computazionale richiesto per risolvere queste equazioni usando il metodo di eliminazione di Gauss è proporzionale a n^3, ma queste equazioni sono sparse, cioè molti coefficienti sono nulli e quindi esistono tecniche anche più efficienti. Questo contrasta con lo sforzo computazionale esponenziale necessario, allo stato attuale delle conoscenze, per decriptare la mappa.

Speriamo che non vi sentiate imbrogliati da questa scoperta. Il processi che hanno luogo nei veri sistemi crittografici sono virtualmente identici a quelli qui descritti, ma le tecniche usate per la codifica sono differenti: non sarebbe stato umanamente possibile fare i calcoli a mano degli algoritmi realmente in uso. Il vero metodo di crittografia a chiave pubblica, che è ancora oggi uno di quelli considerati pù sicuri, è basato sulla difficoltà di fattorizzare numeri molto grandi.

Quali sono i fattori del seguente numero (composto da 100 cifre) 9.412.343.607.359.262.946.971.172.136.294.514.357.528.981.378.983.082.541.347.532.211.942.640.121.301.590.698.6 34.089.611.468.911.681 ? È inutile che impieghiate troppo del vostro tempo a fare tentativi, non li troverete.

Sono 86.759.222.313.428.390.812.218.077.095.850.708.048.977 e 108.488.104.853.637.470.612.961.399.842.972.948.409.834.611.525.790.577.216.753. Non ci sono altri fattori, perché questi due numeri sono primi. Trovare questi fattori dato il loro prodotto è un compito particolarmente arduo: un progetto che impiegherebbe mesi di calcolo per un supercomputer.

In un sistema reale a chiave pubblica Bruno potrebbe usare il numero a 100 cifre come chiave pubblica e i due fattori come chiave privata. La generazione delle chiavi non sarebbe troppo complessa: tutto ciò che

serve è un modo di calcolare numeri primi. Una volta trovati due numeri primi sufficientemente grandi (che non è un problema troppo difficile), si moltiplicano assieme e, oplà, ecco la chiave pubblica. Moltiplicare numeri enormi è un gioco da ragazzi per un computer. Data la chiave pubblica, nessuno potrà trovare quella privata. E se siete preoccupati che qualcuno ci possa riuscire, usate numeri da 200 cifre al posto di quelli da 100 cifre e questo rallenterà di anni il tempo necessario per l'attacco. Nella pratica molto spesso si usano chiavi a 512 bit, che sono grosso modo equivalenti a 155 cifre decimali.

Non abbiamo ancora spiegato come usare i numeri primi per criptare i messaggi usando la chiave pubblica in modo che non si possano decriptare senza la chiave privata. Per fare questo, le cose non sono così semplici come raccontato sopra. Non sono i due numeri primi ad essere utilizzati ma numeri derivati da questi. L'effetto è lo stesso: per violare la codifica occorrerebbe fattorizzare questi numeri. Sarebbe stato possibile tentare di superare le difficoltà e proporre un'attività sulla crittografia a chiave pubblica basata sui numeri primi, ma riteniamo che questo capitolo sia già sufficientemente difficile!

Ma quanto è sicuro un meccanismo basato su numeri primi? La fattorizzazione di numeri molto grandi è stato un problema che ha catturato l'attenzione dei più grandi matematici del mondo per molti secoli e anche se sono stati trovati alcuni metodi per velocizzare la ricerca in modo significativo rispetto alla mera scansione esaustiva delle possibili soluzioni (metodo di forza bruta), nessuno ha trovato un algoritmo realmente veloce, che cioè possa risolvere il problema in tempo polinomiale. Ma dobbiamo fare attenzione. Come abbiamo visto che è possibile violare il codice usato da Bruno senza risolvere il problema della città turistica, ci potrebbero essere modi per violare la codifica basata sui numeri primi senza usare la fattorizzazione. In molti hanno controllato questo metodo e per ora sembra non avere falle.

Un'altra preoccupazione è che ci siano pochi messaggi possibili. Un malintenzionato potrebbe provare a criptarli tutti usando la chiave pubblica del destinatario e confrontare i messaggi intercettati con tutti i messaggi precedentemente criptati. Il metodo usato da Anna evita questo tipo di attacchi perché esistono molti modi diversi di criptare lo stesso messaggio, a seconda di come sono stati scelti i valori che sommati assieme devono dare il valore del messaggio. In pratica i sistemi di crittografia sono sempre progettati in modo che esistano tanti messaggi diversi, anche versioni criptate diverse per lo stesso messaggio, in modo tale che non valga la pena provare a criptarli tutti (e in tutte le varianti di codifica) anche se si disponesse di potentissimi computer.

Non si sa se un metodo efficiente per risolvere la fattorizzazione esista. Nessuno per ora è stato in grado di trovarlo, ma al tempo stesso nessuno è stato in grado di dimostrare che tale metodo non possa

esistere. Se mai un algoritmo efficiente per risolvere il problema della fattorizzazione venisse scoperto, molti dei metodi oggi usati per la crittografia diventerebbero insicuri. Nella parte IV abbiamo discusso i problemi *NP-completi*, che resistono e crollano tutti insieme: se uno di essi diventasse risolubile in modo efficiente allora tutti lo diventerebbero. Dato che molti sforzi sono stati profusi senza successo nella ricerca di questi algoritmi veloci, essi sembrano essere eccellenti candidati per essere alla base di sistemi crittografici sicuri. Ma è difficoltoso usare uno degli algoritmi *NP-completi* e pertanto i progettisti dei sistemi crittografici continuaro a basare i loro metodi su problemi, come la fattorizzazione di un numero, che potrebbero essere più semplici da risolvere degli *NP-completi*, forse molto più semplici. La risposta alla domanda che viene generata da tutto questo vale milioni di euro per il mercato ed è considerata cruciale per la sicurezza nazionale. La crittografia è un'area di ricerca molto attiva nell'ambito dell'informatica.

Per ulteriori approfondimenti

Il libro di Harel, *Algoritmi, lo spirito dell'Informatica* [5], discute la crittografia a chiave pubblica; spiega come usare numeri primi molto grandi per creare un sistema sicuro a chiave pubblica. Un testo molto usato nei corsi di crittografia è *Cryptography and data security* di Dorothy Denning, mentre un altro libro che tratta l'argomento in modo più pratico è *Applied cryptography* di Bruce Schneier [12]. *Turing Omnibus* di Dewdney [3] descrive un altro sistema per realizzare la crittografia a chiave pubblica.