



## 2. Fonaments matemàtics

La criptografia és una disciplina amb un fort contingut matemàtic per diferents motius. Per una banda, la matemàtica permet mesurar de forma precisa la quantitat d'informació que conté un missatge i, per tant, també pot mesurar si quan el xifrem la quantitat d'informació que revela és menor o no en revela cap, de manera que ens dona una mesura de la qualitat del sistema de xifrat que estem utilitzant. D'altra banda, l'explicitació de les funcions de xifrat i desxifrat de les quals hem parlat en l'anterior capítol sovint es realitza utilitzant funcions i expressions matemàtiques. Per aquest motiu, tenir uns bons coneixements de matemàtiques és fonamental per poder entendre el correcte funcionament de la criptografia.

En aquest capítol es proporcionen conceptes bàsics d'aritmètica modular així com algunes propietats dels nombres primers, necessaris en els criptosistemes de clau pública. D'altra banda, tal i com veurem al llarg d'aquesta llibre, els criptosistemes de clau pública així com algunes funcions que s'utilitzen en diferents protocols criptogràfics, basen la seva seguretat en problemes matemàtics difícils de resoldre. És per això que per entendre el grau de seguretat d'aquests sistemes és important comprendre quins són els problemes matemàtics que hi ha al darrere i quina dificultat té la seva resolució. En aquest capítol s'enuncien els problemes matemàtics més utilitzats en criptografia i se'n discuteix la seva complexitat.

Per últim, és important destacar que aquest capítol no pretén en cap cas proporcionar explicacions i demostracions formals dels conceptes matemàtics i menys encara presentar-ne una visió completa. L'objectiu d'aquest capítol és dotar al lector de les eines necessàries per a entendre els criptosistemes i protocols que es descriuran al llarg del llibre. Per a aquells lectors que vulguin aprofundir en les nocions matemàtiques que es presenten en aquest capítol es recomana la lectura de les referències bibliogràfiques que s'indiquen al llarg del text.

## 2.1 Aritmètica modular

Normalment, en la nostra activitat quotidiana treballem amb els nombres reals amb els quals sabem realitzar tot un seguit de càlculs com ara sumes, restes, divisions, multiplicacions, exponenciacions, arrels quadrades, etc. Ara bé, una de les característiques dels nombres reals és que n'hi ha infinits, de manera que la seva representació en un ordinador és impossible. Una possibilitat de resoldre aquest problema és utilitzant conjunts que tinguin un nombre finit d'elements, podent-los representar tots sense cap problema.

L'aritmètica modular és una part de la matemàtica que permet definir tant aquest tipus de conjunts amb un nombre finit d'elements com també les operacions que permeten operar amb els elements d'aquest conjunt, assegurant que l'operació de dos elements del conjunt continuarà proporcionant un altre element del conjunt.

### 2.1.1 Estructures algebraiques: grups, anells i cossos

Des d'un punt de vista informal, podem definir una estructura algebraica com un conjunt d'elements i unes operacions associades que permeten operar amb els elements del conjunt. Depenent de les operacions que definim sobre el conjunt, quantes en definim i quines propietats tinguin, podem classificar l'estructura algebraica en diferents tipus, els més coneguts dels quals són els grups, els anells i els cossos.<sup>1</sup>

Per exemple, si prenem el conjunt dels nombres enters, que es representen per la lletra  $\mathbb{Z}$ , el qual té infinits elements  $\mathbb{Z} = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$  i hi definim l'operació suma tal i com la coneixem, l'estructura algebraica resultant, que podem denotar per  $(\mathbb{Z}, +)$ , és un grup. Això és així donada la següent definició.

**Definició 2.1** Un **grup** és una estructura algebraica en que la operació definida compleix la propietat associativa i, a més, el conjunt sobre el qual està definida l'operació conté l'element neutre i l'element invers d'aquesta operació.

Per exemple, si prenem tres valors enters qualssevol, com ara el  $-3$ , el  $-1$  i el  $2$ , efectivament, veiem que compleixen les propietats anteriors. Per la propietat associativa, tenim que  $((-3) + (-1)) + 2 = (-3) + ((-1) + 2)$ . D'altra banda, l'element neutre de la suma (aquell que sumat amb qualsevol valor dóna ell mateix) pertany als enters, ja que com sabem el neutre de la suma és el  $0$ . L'element invers respecte la suma (aquell que sumat amb un element dóna el neutre de la suma) és el mateix valor canviat de signe, que també pertany al conjunt de nombres enters. Evidentment, aquest exemple concret no és cap demostració que  $(\mathbb{Z}, +)$  és un grup però ens dóna una exemplificació dels conceptes de propietat associativa, element neutre i element invers.

D'altra banda, també podem assegurar que l'estructura algebraica dels nombres naturals amb la suma,  $(\mathbb{N}, +)$  no és un grup ja que si bé la suma sobre els naturals sí que té la propietat associativa i l'element neutre és el  $0$ , que sí que pertany als naturals, l'element invers per la suma de cada element d' $\mathbb{N}$  no pertany a aquest conjunt, ja que els naturals només comprenen nombres positius (i el  $0$ ) i els inversos per la suma d'aquests valors són nombres negatius.

Una altra propietat interessant de les estructures algebraiques és la commutativitat. Un grup

<sup>1</sup>Malgrat que les operacions que es poden definir en una estructura algebraica poden ser tan complicades com es vulguin, a llarg d'aquest text ens restringirem a les dues operacions habituals de suma,  $+$ , i producte,  $\cdot$ , tal i com les coneixem habitualment.

s'anomena commutatiu si l'operació que hi ha definida és commutativa, és a dir, donats dos elements del conjunt  $a$  i  $b$  es compleix que  $a + b = b + a$ .

De la mateixa manera que hem definit una estructura algebraica amb una operació, en podem definir d'altres amb dues operacions diferents. Per exemple, la següent estructura algebraica està formada pels nombres reals amb les operacions de suma i producte:  $(\mathbb{R}, +, \cdot)$ . En aquest cas, podem caracteritzar-les en funció de les propietats que presentin cada una de les operacions, fet que proporciona la definició d'anell i de cos.

**Definició 2.2** Un **anell** és una estructura algebraica amb dues operacions on una d'elles presenta estructura de grup commutatiu, l'altra operació compleix la propietat associativa i a més, amb dues propietats compleixen la distributivitat d'una respecte l'altra.

### Exemple 2.1

L'estructura algebraica  $(\mathbb{Z}, +, \cdot)$  és un anell ja que com hem comentat anteriorment  $(\mathbb{Z}, +)$  és un grup, a més és commutatiu i es compleix la propietat distributiva de la suma respecte el producte, és a dir<sup>a</sup>,  $\forall a, b, c \in \mathbb{Z}, a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ .

<sup>a</sup>Recordeu que el símbol  $\forall$  es llegeix com "per a qualsevol element".

Dels elements d'una estructura algebraica n'hi ha alguns d'especialment rellevants, com ara l'element neutre de cada una de les operacions. Són aquells que operats amb qualsevol element del grup no n'afecten el seu resultat (és a dir, el 0 per a la suma i l'1 per al producte). Com hem vist, en la definició de grup s'exigia que l'element neutre de la suma estigués contingut en el conjunt d'elements. Ara bé, en un anell no s'ha indicat cap condició sobre l'existència o no del neutre del producte. Per tant, podem enunciar la següent definició:

**Definició 2.3** Un **anell amb unitat** és un anell que conté el neutre respecte el producte.

**Exemple 2.2**  $(\mathbb{Z}, +, \cdot)$  és un anell amb unitat ja que  $1 \in \mathbb{Z}$  i 1 és el neutre del producte, perquè compleix que  $\forall a \in \mathbb{Z}, a \cdot 1 = 1 \cdot a = a$ .

L'element unitat en un anell és important perquè ens permet definir el concepte d'element invers.

**Definició 2.4** Donat un anell amb unitat, direm que un element  $a$  és **invertible** si existeix un altre element  $b$  tal que  $a \cdot b = b \cdot a = 1$ , on 1 és l'element unitat.

Amb la definició d'element invertible, ja podem definir l'estructura algebraica més important que hi ha, el cos.

**Definició 2.5** Una estructura algebraica és un **cos**, quan aquesta és un anell amb unitat on qualsevol element, llevat de l'element neutre de la suma, és invertible.

**Notació 2.1.** Utilitzarem l'asterisc per denotar el subconjunt d'elements invertibles. Per exemple,  $\mathbb{Z}^* = \{1, -1\}$  ja que són els únics elements que tenen invers. D'altra banda,  $\mathbb{R}^* = \mathbb{R} \setminus \{0\}$  ja que tots els reals llevat del zero són invertibles.

Així, l'anell  $(\mathbb{Z}, +, \cdot)$  tot i ser un anell amb unitat no és un cos perquè no tots els elements tenen invers. Per exemple, l'invers de 2 és  $\frac{1}{2}$ , i aquesta fracció no pertany als  $\mathbb{Z}$ . De fet, els únics elements

que tenen inversos en  $\mathbb{Z}$  són l'1 i el  $-1$ . D'altra banda, els nombres reals amb la suma i el producte  $(\mathbb{R}, +, \cdot)$  si que són un cos perquè tots els elements, llevat del 0, tenen invers en els  $\mathbb{R}$ .

### 2.1.2 Divisibilitat als enters

Com acabem de veure, els enters amb la suma i el producte són un anell, però no tenen estructura de cos perquè no tots els elements tenen invers respecte el producte. Aquest fet fa que quan calculem una divisió entre dos nombres enters el resultat no sempre sigui un nombre enter. Ara bé, el que si que podem fer és caracteritzar els elements de l'equació resultant d'una divisió entera. Aquesta caracterització la proporciona el següent teorema.<sup>2</sup>

**Teorema 2.1** Donats dos elements  $a \in \mathbb{Z}$  i  $b \in \mathbb{N}$  qualssevol,  $\exists q, r \in \mathbb{Z}$  únics, tals que  $a = b \cdot q + r$ , on  $0 \leq r < b$

El que ens indica aquest teorema és que si dividim l'element  $a$  per un element  $b$  tenim com a resultat un quocient,  $q$ , i un residu,  $r$ . A més, com ja sabem, el residu sempre serà més petit que  $b$  ja que si no ho fos podríem continuar dividint i tindríem un quocient  $q + 1$ , i això ho podríem realitzar de forma repetida fins que el residu sigui més petit que  $b$ .

Una vegada caracteritzats els elements de la divisió entera, podem definir el concepte de divisibilitat als enters.

**Definició 2.6** Donats  $a, b \in \mathbb{Z}$ , diem que  $b$  divideix  $a$  si i només si  $\exists q \in \mathbb{Z}$  tal que  $a = b \cdot q$ . Ho denotarem per  $b|a$ .

Un dels entrebancs que sovint hi ha amb el concepte de divisibilitat és la multiplicitat de definicions que en són equivalents. Així, que l'element  $b$  divideixi a l'element  $a$  és equivalent a dir qualsevol de les següents expressions:

- $b$  és factor d' $a$
- $b$  és divisor d' $a$
- $a$  és divisible per  $b$
- $a$  és múltiple de  $b$

La noció de divisibilitat és important perquè ens permet definir altres eines com ara el màxim comú divisor o caracteritzar alguns nombres, com ara els nombres primers.

**Definició 2.7** Donats dos elements  $a, b \in \mathbb{Z}$  direm que  $d$  és el màxim comú divisor de  $a$  i de  $b$  si  $d$  divideix tant  $a$  com  $b$  i donat qualsevol altre valor  $c$  que també divideixi  $a$  i  $b$ , tenim que  $c < d$ . Denotarem el màxim comú divisor com  $\text{gcd}(a, b) = d$ .

Dels nostres estudis previs en matemàtiques molt probablement el càlcul del màxim comú divisor el sabem fer a partir de la descomposició dels nombres  $a$  i  $b$  en factors primers i prendre'n els comuns amb els menors exponents. És a dir, si volem calcular el  $\text{gcd}(16, 28)$ , com que sabem que  $16 = 2^4 \cdot 1$  i  $28 = 2^2 \cdot 7 \cdot 1$  podem concloure que el  $\text{gcd}(16, 28) = 2^2 = 4$ . Ara bé, aquest sistema per calcular el màxim comú divisor no és gens eficient perquè implica haver de factoritzar els nombres pels quals volem calcular-ne el màxim comú divisor. L'operació de factoritzar, com veurem més endavant en aquest mateix capítol, és un procés molt poc eficient computacionalment, per això

<sup>2</sup>Recordeu que el símbol  $\exists$  es llegeix com "Existeix".

es fa servir l'algorisme d'Euclides que permet calcular el màxim comú divisor de forma eficient, independentment de la mida del nombre. L'algorisme d'Euclides es basa en el següent teorema:

**Teorema 2.2 — Teorema d'Euclides.** Siguin  $a, b, q, r \in \mathbb{Z}$  tals que  $a = b \cdot q + r$  aleshores  $\gcd(a, b) = \gcd(b, r)$ .

Aquest teorema ens indica que podem calcular el màxim comú divisor de dos valors,  $a$  i  $b$ , calculant el màxim comú divisor de dos valors diferents,  $b$  i  $r$ , els quals són tots dos més petits que els anteriors,  $b < a$  i  $r < b$ . Així, calcular el màxim comú divisor consistirà en fer un càlcul recursiu en el que es van dividint els nombres entre ells fins arribar a la condició final, que es concreta en el fet que  $\gcd(x, 0) = x, \forall x \in \mathbb{Z}$ .

### Exemple 2.3 Càlcul de màxim comú divisor utilitzant l'algorisme d'Euclides

Si volem calcular el màxim comú divisor de 2756 i 2621 podem realitzar les següents divisions successives:

$$2756 = 2621 \cdot 1 + 135$$

$$2621 = 135 \cdot 19 + 56$$

$$135 = 56 \cdot 2 + 23$$

$$56 = 23 \cdot 2 + 10$$

$$23 = 10 \cdot 2 + 3$$

$$10 = 3 \cdot 3 + 1$$

$$3 = 1 \cdot 3 + 0$$

D'aquestes divisions i en base al teorema d'Euclides tenim que:

$$\gcd(2756, 2621) = \gcd(2621, 135) = \gcd(135, 56) = \gcd(56, 23) = \gcd(23, 10) = \gcd(10, 3) = \gcd(3, 1)$$

Per tant, com que l'últim residu no nul és el 1, tenim que  $\gcd(2756, 2621) = 1$

**Exercici 2.1** Calcula el màxim comú divisor de 35 i 48.

Una vegada definit el màxim comú divisor de dos nombres podem donar definició de nombres coprimers.

**Definició 2.8** Dos elements  $a$  i  $b$  s'anomenen coprimers quan el  $\gcd(a, b) = 1$ .

Un altre teorema important que ens servirà més endavant per a calcular inversos modulars, és la identitat de Bézout, que permet expressar el màxim comú divisor de dos elements com a combinació lineal dels mateixos.

**Teorema 2.3 — Identitat de Bézout.** Siguin  $a, b \in \mathbb{Z}$  tals que  $\gcd(a, b) = d$  aleshores existeixen uns únics valors  $\lambda, \mu \in \mathbb{Z}$  tals que  $\lambda a + \mu b = d$ .

Tot i que la Identitat de Bézout només ens indica l'existència d'aquests dos valors, podem utilitzar el càlcul del màxim comú divisor amb l'algorisme d'Euclides per calcular-ne exactament els valors  $\lambda$  i  $\mu$ , tal i com es mostra en el següent exemple.

**Exemple 2.4 Càlcul dels coeficients de la identitat de Bézout**

El càlcul dels coeficients de la identitat de Bézout es pot realitzar utilitzant la descomposició que en resulta de l'algorisme d'Euclides. Així si volem calcular el coeficients de la Identitat de Bézout per als valors 2756 i 2621, primer farem el càlcul de les divisions successives de l'algorisme d'Euclides:

$$2756 = 2621 \cdot 1 + 135$$

$$2621 = 135 \cdot 19 + 56$$

$$135 = 56 \cdot 2 + 23$$

$$56 = 23 \cdot 2 + 10$$

$$23 = 10 \cdot 2 + 3$$

$$10 = 3 \cdot 3 + 1$$

Posteriorment, en cada equació n'aïlarem el residu:

$$2756 - (2621 \cdot 1) = 135$$

$$2621 - (135 \cdot 19) = 56$$

$$135 - (56 \cdot 2) = 23$$

$$56 - (23 \cdot 2) = 10$$

$$23 - (10 \cdot 2) = 3$$

$$10 - (3 \cdot 3) = 1$$

i finalment substituïrem en cada equació el valor corresponent per acabar obtenint-ne una sola amb els valors 2756 i 2621:

$$\begin{aligned} 1 &= 10 - (3 \cdot 3) = 10 - ((23 - (10 \cdot 2)) \cdot 3) = (10 \cdot 7) - (23 \cdot 3) = ((56 - (23 \cdot 2)) \cdot 7) - (23 \cdot 3) = \\ &= (56 \cdot 7) - (23 \cdot 17) = (56 \cdot 7) - ((135 - (56 \cdot 2)) \cdot 17) = (56 \cdot 41) - (135 \cdot 17) = ((2621 - (135 \cdot 19)) \cdot 41) - (135 \cdot 17) = \\ &= (2621 \cdot 41) - (135 \cdot 769) = (2621 \cdot 41) - ((2756 - (2621 \cdot 1)) \cdot 796) = \\ &= (2621 \cdot 837) - (2756 \cdot 796) \end{aligned}$$

Així, tenim que

$$1 = 2621 \cdot 837 + 2756 \cdot (-796)$$

i per tant els coeficients de la identitat de Bézout per a 2756 i 2621 són  $-796$  i  $837$  respectivament.

**Exercici 2.2** Calcula els coeficients de la indentitat de Bezout de 35 i 48.

Com ja hem indicat anteriorment, a més del màxim comú divisor, el concepte de divisibilitat també ens permet definir els nombres primers.

**Definició 2.9** Direm que un nombre  $p \in \mathbb{N}$ , amb  $p \neq 0$ , és primer si només és divisible per ell mateix i per 1.

L'últim concepte relacionat amb la divisibilitat als enters que definirem és la funció  $\phi$  d'Euler, la qual, com veurem més endavant, és la base del funcionament de l'algorisme de xifrat RSA.

**Definició 2.10 — Funció  $\phi$  d'Euler.** La funció  $\phi$  d'Euler d'un valor natural  $n$ ,  $\phi(n)$ , es defineix com el cardinal del conjunt de nombres coprimers amb  $n$  més petits que  $n$ . És a dir:

$$\phi(n) = \#\{a, 0 \leq a < n, \text{ tal que } \gcd(a, n) = 1\}$$

**Exemple 2.5 Càlcul de funció fi d'Euler**

Si volem calcular el valor de  $\phi(10)$  seguint la definició d'aquesta funció, ens caldrà calcular tots els nombres més petits que 10 que són coprimers amb 10, és a dir, que el  $\gcd(x, 10) = 1$ . Si els calculem resultarà que són els següents  $\{1, 3, 7, 9\}$ , per tant, el valor de la funció fi d'Euler serà el nombre d'elements d'aquest conjunt, és a dir 4.

$$\phi(10) = \#\{1, 3, 7, 9\} = 4$$

Més enllà de calcular tots els elements coprimers amb  $n$  i comptar-los hi ha tècniques més eficients per calcular la funció fi d'Euler. La més eficient que es coneix consisteix a descompondre el valor  $n$  en factors primers. Un cop descompost el valor, s'utilitzen les següents propietats:

- Si  $p$  és un nombre primer,  $\phi(p) = p - 1$ . Això és fàcil de veure perquè tots els elements més petits que  $p$  seran coprimers amb  $p$  donat que no tenen cap factor en comú, justament pel fet que  $p$  és primer.
- Si  $n = p \cdot q$  tals que  $p$  i  $q$  són coprimers, aleshores  $\phi(n) = \phi(p) \cdot \phi(q)$ .
- Si  $n$  és una potència d'un primer,  $n = p^k$ , aleshores  $\phi(p^k) = p^k - p^{k-1} = p^{k-1}(p - 1)$

Com podem veure, malgrat que aquesta sigui la millor manera de calcular la funció d'Euler, donat que implica descompondre el valor en factors primers, no és una tasca computacionalment eficient quan el valor del nombre és molt elevat.

**Exercici 2.3** Calcula quin és el valor de  $\phi(527)$ .

**2.1.3 Aritmètica modular amb enters**

En el primer apartat d'aquest capítol hem vist quines són les propietats que ha de tenir una estructura algebraica per a ser un grup, un anell o un cos. Com ja hem comentat, els cossos són estructures algebraiques molt versàtils gràcies a les propietats que presenten les seves operacions. Els exemples d'anells o cossos que hem vist en l'apartat anterior, i els que coneixem normalment, són exemples on el conjunt d'elements és un conjunt infinit. Així, el conjunt dels enters amb la suma i el producte  $(\mathbb{Z}, +, \cdot)$  és un anell, però els enters és un conjunt d'elements infinit. Igualment, el conjunt dels reals amb la suma i el producte  $(\mathbb{R}, +, \cdot)$  és un cos, però, de nou, els reals són un conjunt infinit. Per tant, ens podem preguntar si podem crear estructures algebraiques que siguin anells o cossos, però que tinguin un nombre finit d'elements. I la resposta a aquesta pregunta és afirmativa.

**Definició 2.11** Definirem el conjunt dels enters mòdul  $n$ , per a  $n \geq 2$ , i el denotarem per  $\mathbb{Z}_n$ , com tots els nombres enters entre 0 i  $n - 1$ , és a dir  $\mathbb{Z}_n = \{0, 1, \dots, n - 1\}$ .

Com és evident, els enters mòdul  $n$  és un conjunt finit, ja que conté exactament  $n$  elements. Per tant, si aconseguim definir una operació suma i una operació producte que tinguin les propietats que hem enumerat en anteriors apartats, això ens permetrà construir anells i cossos amb un nombre finit d'elements. Per definir tant la suma com el producte a  $\mathbb{Z}_n$  utilitzarem la definició de suma i producte d'enters que ja coneixem. Ara bé, caldrà anar en compte perquè és important que les operacions siguin operacions internes, és a dir, quan operem dos elements d'un conjunt cal que el



resultat sigui un element del mateix conjunt. Això amb els conjunts en els que estem acostumats a treballar ja passa, perquè si sumem (o multipliquem) dos elements enters en dóna un enter i si sumem (o multipliquem) dos nombres reals, el resultat també és un nombre real. Ara bé, si prenem la suma tal i com la coneixem i considerem ara el conjunt  $\mathbb{Z}_5 = \{0, 1, 2, 3, 4\}$  veurem que cal anar amb compte perquè si fem la suma de dos elements de  $\mathbb{Z}_5$ , per exemple  $4 + 4$  el resultat és 8, que no és un element del conjunt  $\mathbb{Z}_5$ . Per tal de resoldre aquest problema el que farem és pensar  $\mathbb{Z}_n$  com una reducció de tots els enters. Si tenim una manera per “reduir” qualsevol enter a un valor de  $\mathbb{Z}_n$  ja haurem aconseguit l’objectiu, perquè per una banda tenim definides la suma i el producte de manera que el resultat és un enter, i amb l’eina de reducció, podríem reduir el resultat de l’operació a un element de  $\mathbb{Z}_n$ .

Per obtenir aquesta funció de reducció de tots els enters a  $\mathbb{Z}_n$  només ens cal recuperar el teorema de divisió entera que hem vist anteriorment. Efectivament, si volem reduir un element enter  $a$  a un enter entre 0 i  $n - 1$  només cal fer la divisió entera de  $a$  entre  $n$ . Aquesta divisió tindrà un residu únic. A més, com que hem dividit per  $n$ , aquest residu serà un valor entre 0 i  $n - 1$  que és justament el que ens interessa.

Si tornem ara a l’exemple dels enters mòdul 5, que recordem que és el conjunt format per  $\mathbb{Z}_5 = \{0, 1, 2, 3, 4\}$ , veiem que la suma que proposàvem  $4 + 4$  donava com a resultat 8. Ara bé, si reduïm el 8 tal i com hem descrit anteriorment, tenim que la divisió entera de 8 entre 5 dóna com a quocient 1 i de reste 3. Per tant, podem concloure que el 8 equival a un 3 a  $\mathbb{Z}_5$  i que per tant, la suma que teníem ens queda  $4 + 4 = 8 = 3$  a  $\mathbb{Z}_5$ . De la mateixa manera que hem pogut definir la suma, podem fer el mateix amb el producte. Així  $3 \cdot 4 = 2 \pmod{5}$  ja que si dividim 12 (que és el resultat de 3 per 4) entre 5 tenim 2 de residu.<sup>3</sup>

#### Exercici 2.4 Quants elements té el conjunt $\mathbb{Z}_{25}$ ?

Una vegada definit el conjunt dels enters mòdul  $n$  i les operacions de suma i producte dins d’aquest conjunt, ja podem enunciar el següent teorema:

**Teorema 2.4** L’estructura algebraica  $(\mathbb{Z}_n, +, \cdot)$  amb la suma i el producte tal i com els hem definit anteriorment i per a qualsevol valor  $n \geq 2$  és un anell commutatiu amb unitat.

D’aquesta manera, hem pogut definir un anell sobre un conjunt d’elements finits, com és el cas de  $\mathbb{Z}_n$ . Per fer operacions de suma i producte a  $\mathbb{Z}_n$  ens caldrà únicament operar de forma normal amb els enters i un cop obtingut el resultat final reduir-lo al mòdul on treballem. A més, aquesta reducció al mòdul la podem fer al final dels càlculs o en qualsevol moment, per exemple, per simplificar els valors amb els que estem treballant.

#### Exemple 2.6 Càlculs en anells modulars

Si volem saber el valor de l’expressió  $5 \cdot (4 + 14) - 3 \cdot 8$  a  $\mathbb{Z}_{10}$  podem fer el següents càlculs:

$$5 \cdot (4 + 14) - 3 \cdot 8 \pmod{10}$$

$$5 \cdot (18) - 24 \pmod{10}$$

$$90 - 24 \pmod{10}$$

<sup>3</sup>Les equacions a  $\mathbb{Z}_n$  s’anomenen equacions modulars. Per indicar l’equació modular  $4 + 4$  a  $\mathbb{Z}_5$  escriurem el següent:  $4 + 4 = 3 \pmod{5}$ .



$$66 \pmod{10}$$

$$6 \pmod{10}$$

on l'últim pas prové del fet que el reste de dividir 66 entre 10 és 6.

Fixeu-vos que una altra manera de realitzar el càlcul és que en el primer pas, haguéssim reduït tant el 18 com el 24 mòdul 10, això és hagués proporcionat l'expressió

$$5 \cdot (8) - 4 \pmod{10}$$

més fàcil de gestionar per la mida dels valors. Aquesta expressió també hagués proporcionat el mateix resultat final, ja que

$$5 \cdot (8) - 4 \pmod{10} = 40 - 4 \pmod{10} = 36 \pmod{10} = 6 \pmod{10}$$

Fixeu-vos que fins ara només ens hem referit a operar amb sumes, restes i multiplicacions. Això és així perquè, fins al moment, hem pogut assegurar que l'estructura algebraica que hem construït és un anell. Però en un anell no necessàriament tots els elements tenen invers pel producte. Arribats a aquest punt, ens podem preguntar si l'estructura algebraica  $(\mathbb{Z}_n, +, \cdot)$ , a més d'un anell és també un cos.

Si assumim el Teorema 2.4 com a cert, és a dir que  $(\mathbb{Z}_n, +, \cdot)$  és un anell commutatiu amb unitat, per veure si  $(\mathbb{Z}_n, +, \cdot)$  és un cos, atenent-nos a la definició de cos que hem donat, només ens cal comprovar dos fets. El primer és que aquest anell té unitat. I el segon, que tot element de l'anell, llevat del neutre de la suma, és invertible. La primera comprovació és trivial, ja que sabem que l'element 1, que és la unitat del producte, sempre pertany a  $\mathbb{Z}_n$  (ja que hem dit que  $n \geq 2$ ). Ara bé, la segona propietat no sempre és certa, i dependrà del tipus de valor  $n$ .

**Teorema 2.5** L'estructura algebraica  $(\mathbb{Z}_p, +, \cdot)$  és un cos si i només si el valor  $p$  és un nombre primer.

Aquest teorema ens indica que, per exemple, l'estructura algebraica  $(\mathbb{Z}_{17}, +, \cdot)$  és un cos perquè 17 és un nombre primer. Per tant, qualsevol element a  $\mathbb{Z}_{17}$ , que recordem que està format pels elements  $\{0, 1, 2, 3, 4, \dots, 15, 16\}$ , té invers pel producte. El fet que qualsevol element tingui invers és molt rellevant perquè permet fer divisions amb elements d'aquest conjunt. En efecte, si volem calcular  $\frac{2}{3}$  només hem de saber quan val l'invers de 3, és a dir  $\frac{1}{3}$  i multiplicar aquest valor per 2.

### Inversos modulars

Com acabem de veure, l'últim teorema de l'apartat anterior ens indica que l'estructura algebraica  $(\mathbb{Z}_p, +, \cdot)$  és un cos si  $p$  és primer. Per tant, sabem que per a qualsevol element de  $\mathbb{Z}_p$ , llevat del zero, podem calcular-ne el seu valor invers. Vegem com fer-ho.

En primer lloc, és important recordar la definició d'element invers. Per exemple, si volem calcular l'invers de 3 a  $\mathbb{Z}_{17}$  sabem que estem buscant un valor que multiplicat per 3 valgui 1 a  $\mathbb{Z}_{17}$ . A més, com que sabem que  $\mathbb{Z}_{17}$  és un cos sabem que l'invers de 3 ha de pertànyer a  $\mathbb{Z}_{17}$ , per tant ha de ser un valor del conjunt  $\{0, 1, 2, 3, 4, \dots, 15, 16\}$ . Si multipliquem cada un d'aquests elements d'aquest conjunt per 3, un dels productes ens donarà 1. En efecte, si fem el producte  $3 \cdot 6 = 18$  veiem que el resultat, reduït a mòdul 17, és 1. Així doncs, l'invers de 3 mòdul 17 serà 6.

Evidentment, aquest sistema que acabem de descriure no és bo per calcular inversos modulars en cas que el valor de  $p$  sigui molt gran, ja que ens requeriria fer molts productes. Una manera eficient

de calcular inversos és utilitzant la identitat de Bézout.

Fixeu-vos que si volem calcular l'invers de  $x \in \mathbb{Z}_p$ , on  $p$  és primer, com que  $x$  és més petit que  $p$  i  $p$  és primer, tenim que  $\gcd(x, p) = 1$ , ja que si el màxim comú divisor  $d$  no fos 1,  $p$  no seria primer perquè es podria dividir per  $d$ . Ara bé, si  $\gcd(x, p) = 1$ , per la Identitat de Bézout, sabem que existeixen dos elements  $\lambda$  i  $\mu$  tals que  $x \cdot \lambda + p \cdot \mu = 1$ . Però fixeu-vos que si aquesta equació l'expresssem modularment a  $\mathbb{Z}_p$  ens queda  $x \cdot \lambda + p \cdot \mu = 1 \pmod{p}$  i si la reduïm modularment obtenim l'equació equivalent  $x \cdot \lambda = 1 \pmod{p}$  donat que  $p$  a  $\mathbb{Z}_p$  val 0 (ja que el reste de dividir  $p$  entre  $p$  és 0). Per tant, el valor que multiplicat per  $x$  dona 1 mòdul  $p$  és justament  $\lambda$ . Dit en altres paraules, un dels coeficients de la identitat de Bézout és el que ens proporciona l'invers modular.

### Exemple 2.7 Càlcul d'invers modular

Si volem calcular l'invers de 9 mòdul 11 calcularem els coeficients de la identitat de Bézout tal i com hem mostrat en exemples anteriors, utilitzant l'algorisme de les divisions successives:

$$11 = 9 \cdot 1 + 2$$

$$9 = 2 \cdot 4 + 1$$

De la segona equació tenim  $1 = 9 - (2 \cdot 4)$  i de la primera equació  $2 = 11 - (9 \cdot 1)$ . Si les combinem ens queda:  $1 = 9 - (11 - (9 \cdot 1) \cdot 4) = 9 - (11 \cdot 4) + (9 \cdot 4) = 5 \cdot 9 - 11 \cdot 4$ . Per tant, els coeficients de la identitat de Bézout de 9 i 11 són 5 i  $-4$  respectivament ja que  $\gcd(9, 11) = 1 = 5 \cdot 9 + 11 \cdot (-4)$ . Per tant, si reduïm aquesta equació mòdul 11 ens queda  $5 \cdot 9 \pmod{11}$ , és a dir, l'invers de 9 mòdul 11 és 5. Això és fàcil de comprovar, perquè  $9 \cdot 5 = 45$  i el reste de dividir 45 per 11 és, efectivament, 1.

### Exercici 2.5 Troba l'invers de 7 a $\mathbb{Z}_{37}$

### Exercici 2.6 Realitza els següents càlculs a $\mathbb{Z}_{37}$

- $20 + 20$
- $20 \cdot 4$
- $20^2$
- $\frac{20}{7}$

### Exercici 2.7 Per què l'estructura algebraica $(\mathbb{Z}_{37}, +, \cdot)$ és un cos?

## El teorema d'Euler

L'aritmètica modular conté innombrables resultats, que tot i ser molt interessants queden fóra de l'abast d'aquest llibre. En aquest apartat ens centrarem únicament amb el teorema d'Euler, que és la pedra angular del funcionament de l'algorisme de clau pública RSA.

**Teorema 2.6 — Teorema d'Euler.** Sigui  $n$  un nombre natural i  $\phi(n)$  la seva funció fi d'Euler. Si  $\gcd(x, n) = 1$ , aleshores:

$$x^{\phi(n)} = 1 \pmod{n}$$

Aquest teorema ens indica que quan estem en un anell modular, qualsevol valor elevat a la funció d'Euler del mòdul és igual a la identitat. La importància d'aquest teorema en l'RSA és que permet

demostrar, com veurem més endavant, que quan xifrem un missatge i posteriorment el desxifrem el resultat que obtenim és el text en clar original.

Una altra implicació del teorema d'Euclides és el següent resultat que ens permet calcular inversos modulars per mitjà d'exponenciacions:

**Proposició 2.1** Sigui  $x \in \mathbb{Z}_n$ , l'invers d' $x$  a  $\mathbb{Z}_n$  és  $x^{\phi(n)-1}$ .

**Demostració:** La demostració d'aquesta proposició és immediata fent servir el teorema d'Euclides. Efectivament si multipliquem  $x$  per  $x^{\phi(n)-1}$  tenim  $x^{\phi(n)}$  que val 1 a  $\mathbb{Z}_n$  pel teorema d'Euclides, cosa que prova que són inversos. ■

### Exemple 2.8 Càlcul d'invers modular

Si volem calcular l'invers de 2 a  $\mathbb{Z}_{11}$  podem calcular la funció d'Euler del mòdul  $\phi(11) = 10$  i posteriorment calcular la següent potència:

$$2^{10-1} = 2^9 = 512 = 6 \pmod{11}$$

Per tant, l'invers de 2 mòdul 11 val 6.

### Elements primitius

Quan treballem amb cossos finits, el fet que el conjunt d'elements que tenim sigui finit junt amb el fet que les operacions entre elements han de ser internes, ens trobem en situacions que no es donen quan treballem amb conjunts de mida infinita. Un exemple d'aquest cas el trobem en la definició d'ordre d'un element.

**Definició 2.12** L'ordre d'un element  $a \in \mathbb{Z}_n$  és el mínim exponent  $i \in \mathbb{N}; i > 0$  tal que  $a^i = 1$  mòdul  $n$ .

### Exemple 2.9 Ordre d'un element

L'ordre de l'element 5 a  $\mathbb{Z}_{42}$  és 6 ja que  $5^1 = 5$ ,  $5^2 = 25$ ,  $5^3 = 41$ ,  $5^4 = 37$ ,  $5^5 = 17$ ,  $5^6 = 1$ .

Una vegada definit el concepte d'ordre, podem definir el concepte d'element primitiu.

**Definició 2.13** Un element  $g \in \mathbb{Z}_n$  és un **element primitiu** si té ordre  $\phi(n)$ .

Fixeu-vos que en el cas que el mòdul del nostre conjunt sigui un nombre primer  $p$ , tenim que l'element  $g$  serà primitiu a  $\mathbb{Z}_p$  si té ordre  $\phi(p)$ . Com que  $p$  es primer sabem que  $\phi(p) = p - 1$ , per tant, l'ordre de  $g$  serà  $p - 1$ . Això vol dir que si prenem  $g$  i anem calculant potències ens generarem  $p - 1$  elements diferents. Ara bé, fixeu-vos que  $\mathbb{Z}_p$  té  $p$  elements diferents i, en concret, llevat del 0 tots són invertibles (ja que  $\mathbb{Z}_p$  és un cos). Per tant,  $\mathbb{Z}_p^*$  té  $p - 1$  elements i podem concloure que les  $p - 1$  potències diferents de  $g$  generen tots els elements invertibles de  $\mathbb{Z}_p$ . Per exemple, si prenem  $\mathbb{Z}_7 = \{0, 1, 2, 3, 4, 5, 6\}$ , veiem que els seus elements invertibles són  $\mathbb{Z}_7^* = \{1, 2, 3, 4, 5, 6\}$ . Tenint en compte que 3 és un element primitiu, podem comprovar que les seves potències poden generar tots els elements invertibles:  $3^1 = 3$ ,  $3^2 = 2$ ,  $3^3 = 6$ ,  $3^4 = 4$ ,  $3^5 = 5$ ,  $3^6 = 1$ .

### Sistemes d'equacions modulars

En apartats anteriors hem vist la resolució d'equacions modulars utilitzant els mecanismes estàndards de resolució d'equacions però tenint en compte que el conjunt on treballem és  $\mathbb{Z}_n$ . Ara bé, es pot donar el cas que ens interressi resoldre un sistema d'equacions definides sobre diferents mòduls, com per exemple:

$$\begin{cases} 3x + 5 = 0 & (\text{mod } 11) \\ 3x - 2 = 0 & (\text{mod } 5) \end{cases}$$

En aquest cas, el teorema xinès dels residus ens proporciona informació sobre l'existència d'una solució.

**Teorema 2.7 — Teorema xinès dels residus.** Siguin  $n_1$  i  $n_2$  dos elements naturals tals que  $\text{mcd}(n_1, n_2) = 1$  aleshores, el sistema d'equacions modulars:

$$\begin{cases} x = a_1 & (\text{mod } n_1) \\ x = a_2 & (\text{mod } n_2) \end{cases}$$

té una única solució mòdul  $n = n_1 \cdot n_2$  definida per l'equació:

$$x = \lambda \cdot n_2 \cdot a_1 + \mu \cdot n_1 \cdot a_2 \pmod{n}$$

on  $\lambda$  i  $\mu$  són els coeficients de la identitat de Bézout  $\mu \cdot n_1 + \lambda \cdot n_2 = 1$ .

#### Exemple 2.10 Resolució d'un sistema d'equacions modular

Si volem resoldre el sistema d'equacions següent:

$$\begin{cases} 3x + 5 = 0 & (\text{mod } 11) \\ 3x - 2 = 0 & (\text{mod } 5) \end{cases}$$

En primer lloc ens cal expressar les equacions en el format adequat. És a dir:

$$\begin{cases} x = 2 & (\text{mod } 11) \\ x = 4 & (\text{mod } 5) \end{cases}$$

A continuació, calcularem els elements de la identitat de Bézout de 11 i 5, que són  $\mu = 1$  i  $\lambda = -2$ , ja que  $1 \cdot 11 + -2 \cdot 5 = 1$ . Per tant, la solució serà:

$$x = \lambda \cdot n_2 \cdot a_1 + \mu \cdot n_1 \cdot a_2 \pmod{n} = -2 \cdot 5 \cdot 2 + 1 \cdot 11 \cdot 4 \pmod{55} = 24 \pmod{55}$$

### Residus quadràtics i arrels quadrades modulars

Quan treballem amb estructures algebraiques finites, una de les operacions que també ens interessarà realitzar són arrels quadrades. Com veurem en aquest apartat, no tots els elements tindran arrels quadrades i, a més, en cas que en tinguin en poden tenir més de dues.

**Definició 2.14** Sigui  $p$  un nombre primer. Direm que  $y \in \mathbb{Z}_p$  és un residu quadràtic si existeix un valor  $x \in \mathbb{Z}_p$  tal que  $x^2 = y$ . En cas que no existeixi aquest valor, direm que  $y$  no és un residu quadràtic.

En el cos  $\mathbb{Z}_p$ , hi ha el mateix nombre d'elements que són residu quadràtic que elements que no ho són, és a dir un total de  $\frac{p-1}{2}$  elements. A més, hi ha una manera fàcil de calcular si un element és un residu quadràtic aplicant la següent expressió:

$$y^{\frac{p-1}{2}} = \begin{cases} 1 & \text{si } y \text{ és un residu quadràtic} \\ -1 & \text{si } y \text{ no és un residu quadràtic} \end{cases}$$

Un cop sabem que un valor és un residu quadràtic, podem calcular-ne les seves arrels quadrades, ja que sabem que existeixen. Amb la fórmula anterior, és molt simple comprovar si un valor és un residu quadràtic o no ho és. Ara bé, calcular-ne les arrels quadrades suposa una mica més de feina. De tota manera, per a nosaltres ens serà suficient saber que existeixen algorismes eficients<sup>4</sup> que poden calcular arrels quadrades d'un residu quadràtic a  $\mathbb{Z}_p$  encara que el valor  $p$  sigui molt gran, això sí, sempre que  $p$  sigui un nombre primer.

Quan deixem els nombres primers com a base de la nostra estructura algebraica i adoptem elements que són producte de primers, les coses es compliquen.

**Proposició 2.2** Siguin  $p$  i  $q$  dos nombres primers i  $n$  el seu producte,  $n = p \cdot q$ . Aleshores, a  $\mathbb{Z}_n$  hi ha exactament  $\frac{\phi(n)}{4}$  residus quadràtics i cada un d'ells té exactament quatre arrels quadrades.

Un punt important a tenir en compte és que si un element  $y$  és residu quadràtic a  $\mathbb{Z}_n$  i  $n = p \cdot q$ , aleshores  $y$  també és residu quadràtic a  $\mathbb{Z}_p$  i a  $\mathbb{Z}_q$ . Aquest fet ens proporciona un sistema per calcular arrels quadrades d'un residu quadràtic  $y$  a  $\mathbb{Z}_n$ , ja que per calcular-les només ens caldrà calcular les arrels quadrades de  $y$  a  $\mathbb{Z}_p$  i a  $\mathbb{Z}_q$  i combinar-les.

### Exemple 2.11 Càlcul d'arrels quadrades a $\mathbb{Z}_n$ amb $n = p \cdot q$ producte de dos primers

Calculem les arrels quadrades de 4 a  $\mathbb{Z}_{15}$ .

Donat que 15 és producte de dos primers, sabem que 4 té 4 arrels quadrades a  $\mathbb{Z}_{15}$ , que denotarem per  $x_1, x_2, x_3, x_4$ . Per calcular-les, calcularem primer les arrels quadrades de 4 a  $\mathbb{Z}_3$  i a  $\mathbb{Z}_5$ .

Aquest cas és molt simple perquè sabem que, en els reals, les arrels de 4 són 2 i -2. Per tant, les arrels quadrades de 4 a  $\mathbb{Z}_3$  seran els valors  $y_1 = 2, y_2 = 1$  i les arrels quadrades de 4 a  $\mathbb{Z}_5$  seran els valors  $z_1 = 2, z_2 = 3$

Ara bé, sabem que les arrels quadrades de 4 a  $\mathbb{Z}_{15}$  que busquem també ho han de ser a  $\mathbb{Z}_3$  i a  $\mathbb{Z}_5$ . Això fa que puguem plantejar el següent sistema d'equacions:

<sup>4</sup>L'algorisme de Tonelli-Shank permet, en temps polinomial, calcular arrels quadrades d'un residu quadràtic a  $\mathbb{Z}_p$ , per a qualsevol valor  $p$  primer.

$$\begin{cases} x = y_i \pmod{3} \\ x = z_i \pmod{5} \end{cases}$$

Si ens hi fixem, aquí tenim un sistema d'equacions modulars que podem resoldre amb el teorema xinès del residu, tal i com hem explicat en l'apartat anterior. En aquest cas, com que els coeficients de la Identitat de Bezout per 3 i 5 valen 2 i  $-1$ , respectivament, ja que  $1 = 2 \cdot 3 - 1 \cdot 5$ , la solució del sistema amb el teorema xinès del residu ens queda:

$$m = 2 \cdot 3 \cdot z_i - 1 \cdot 5 \cdot y_i$$

Per tant, només ens queda substituir els valors de  $y_i$  i  $z_i$  per  $i = \{1, 2\}$  per trobar les quatre arrels quadrades de 4 a  $\mathbb{Z}_{15}$  que seran  $\{7, 2, 13, 8\}$ .

Hem vist que decidir si un element és un residu quadràtic a  $\mathbb{Z}_p$  és fàcil en el cas que  $p$  sigui un nombre primer. Ara bé, decidir-ho a  $\mathbb{Z}_n$  amb  $n$  producte de dos primers és un problema computacionalment intractable. Com també ho és el càlcul de les arrels quadrades. Si ens fixem amb l'exemple anterior, per calcular les arrels quadrades  $x_i$  hem hagut de calcular primer les  $y_i$  i les  $z_i$  per posteriorment combinar-les. Ara bé, això ho hem pogut fer perquè coneixíem la factorització del mòdul, en aquest cas sabíem que  $n = 3 \cdot 5$ . Ara bé, si no coneixem la descomposició del mòdul, no podem calcular les arrels quadrades. De fet, el següent teorema mostra l'equivalència del càlcul d'arrels quadrades i la factorització del mòdul.

**Teorema 2.8** Sigui  $n = p \cdot q$ , on  $p$  i  $q$  són primers imparells diferents. Si  $x$  i  $y$  són arrels quadrades essencialment diferents d'un element de  $\mathbb{Z}_n$  aleshores  $\gcd(x + y, n)$  és un dels dos factors  $p$  o  $q$ .

Dit d'una altra manera, el teorema anterior ens indica que saber calcular arrels quadrades a  $\mathbb{Z}_n$  és el mateix que saber calcular la factorització del nombre  $n$ .

### Exemple 2.12 Equivalència entre arrels quadrades i factorització

Suposem que volem calcular la factorització del valor  $n = 925219$  però no tenim un algorisme per factoritzar-lo. D'altra banda, sabem que les quatre arrels quadrades de 524422 a  $\mathbb{Z}_{925219}$  valen  $\{272635, 402576, 522643, 652584\}$ .

Si prenem dues d'aquestes arrels quadrades que siguin essencialment diferents, per exemple,  $\{272635, 402576\}$ , i calculem  $\gcd(272635 + 402576, 925219) = \gcd(675211, 925219) = 947$  podem veure que efectivament 947 és un dels primers que formen el valor 925219 ja que si els dividim  $\frac{925219}{947} = 977$  la seva divisió és exacta i ens proporciona l'altre factor primer:  $925219 = 947 \cdot 977$ .

El concepte arrels essencialment diferents fa referència al fet que les dues arrels no poden ser l'inversa una de l'altra. És a dir, si ens hi fixem, per exemple, les arrels  $\{272635, 652584\}$  compleixen que  $652584 = -272635 \pmod{925219}$ . Això fa que si haguéssim agafat aquestes dues arrels per fer el càlcul del màxim comú divisor no haguéssim aconseguit cap resultat ja que

$$652584 + 272635 = 0 \pmod{925219}.$$

Noteu que hem pogut factoritzar un valor “únicament” calculant un màxim comú divisor, una operació que és computacionalment simple fent servir l’Algorisme d’Euclides. El truc ha estat que tenim totes les arrels quadrades d’un element.

### 2.1.4 Aritmètica modular amb polinomis

En els apartats anteriors hem vist una manera de construir cossos finits, en concret cossos finits que tinguin un nombre primer d’elements. La pregunta que ens podríem fer ara és si podem crear cossos finits que la quantitat total d’elements no sigui un nombre primer. La resposta a aquesta pregunta és afirmativa i a continuació veurem com és possible crear cossos finits que el nombre total d’elements sigui una potència d’un nombre primer.

Quan en l’apartat anterior hem parlat d’estructures algebraiques no hem fet menció d’una altra estructura que algebraica força coneguda que també és una anell. Aquesta estructura és l’anell de polinomis amb coeficients als reals, que denotarem per  $(\mathbb{R}[x], +, \cdot)$ . Com ja sabem, un element  $a(x) \in \mathbb{R}[x]$  és un element del tipus  $a(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \dots$  on  $a_i \in \mathbb{R}$ . Amb els polinomis sabem perfectament sumar-los (sumant les components del mateix grau) i també multiplicar-los (component a component). A més, igual com teníem amb la divisió dels enters, en el cas que el grau d’un polinomi  $a(x)$  sigui més gran que el grau del polinomi  $b(x)$  també podem dividir el polinomi  $a(x)$  entre el polinomi  $b(x)$  i obtindrem dos polinomis  $q(x)$  i  $r(x)$  on es complirà que  $a(x) = b(x) \cdot q(x) + r(x)$  i a més el grau de  $r(x)$  és més petit estricte que el grau de  $b(x)$ .

#### Exemple 2.13 Operacions bàsiques amb polinomis

Donats els polinomis:

$$a(x) = 3 + \frac{1}{2}x + 2x^2$$

$$b(x) = 1 + x$$

Suma de dos polinomis:

$$a(x) + b(x) = 3 + \frac{1}{2}x + 2x^2 + 1 + x = 4 + \frac{3}{2}x + 2x^2$$

Producte de dos polinomis:

$$a(x) \cdot b(x) = (3 + \frac{1}{2}x + 2x^2) \cdot (1 + x) = 3 + \frac{7}{2}x + \frac{5}{2}x^2 + 2x^3$$

Divisió de polinomis:

$$a(x) = q(x) \cdot b(x) + r(x) = (2x - \frac{3}{2}) \cdot (x + 1) + \frac{9}{2}$$

Amb els anells de polinomis podem definir algorismes anàlegs als que hem vist per als enters. Així podrem calcular el màxim comú divisor de dos polinomis o els coeficients de la identitat de Bézout.

Com és evident, el nombre d’elements de  $(\mathbb{R}[x], +, \cdot)$  és infinit ja que el nombre d’elements d’ $\mathbb{R}$  ho és. Ara bé, podríem intentar limitar el nombre de coeficients a utilitzar canviant el conjunt  $\mathbb{R}$  pel conjunt  $\mathbb{Z}_n$ . D’aquesta manera, l’estructura algebraica  $(\mathbb{Z}_n[x], +, \cdot)$  tindria “menys” elements que



$(\mathbb{R}[x], +, \cdot)$ . Malgrat tot,  $(\mathbb{Z}_n[x], +, \cdot)$  continua tenint infinits elements perquè tot i que hem limitat el nombre de coeficients que podem triar en el polinomi (ara només poden ser enters entre el 0 i  $n - 1$ ) continuem tenint el grau del polinomi il·limitat. Per tant, ens cal que el nostre conjunt, a més de tenir el nombre de coeficients limitat, també tingui el grau del polinomi limitat.

Per limitar el grau del polinomi podem utilitzar un mecanisme anàleg al que hem fet als enters. Prenem tots els polinomis de  $\mathbb{Z}_n[x]$  i els dividim per un polinomi de grau fixat  $k$ . Com que la divisió d'un polinomi de grau qualsevol per un polinomi de grau  $k$  sempre ens donarà de reste un polinomi de grau més petit o igual que  $k - 1$ , si prenem tots els restes d'aquesta divisió haurem aconseguit especificar tots els polinomis de  $\mathbb{Z}_n[x]$  que tenen grau com a molt  $n - 1$ , és a dir elements del tipus  $a(x) = a_0 + a_1x + a_2x^2 + \dots + a_{k-1}x^{k-1}$  on  $a_i \in \mathbb{Z}_n$ . Si ens hi fixem, aquí sí que el nombre total d'elements d'aquest conjunt és un nombre finit, i concretament valdrà  $n^k$ . Aquest conjunt el denotarem com  $\mathbb{Z}_n[x]/a(x)$  i seran els polinomis amb coeficients a  $\mathbb{Z}_n$  mòdul el polinomi  $a(x)$ .

Així doncs, podem prendre el conjunt  $\mathbb{Z}_n[x]/a(x)$  i definir-hi una suma i un producte estàndards de polinomis. Si volem que les operacions siguin internes, és a dir que la suma i productes d'elements de  $\mathbb{Z}_n[x]/a(x)$  continuïn estant en el mateix conjunt haurem de fer el mateix que fèiem en els enters, és a dir reduir el resultat modularment.

**Teorema 2.9** Donat un nombre  $p$  primer i un polinomi  $m(x) \in \mathbb{Z}_p$ , l'estructura algebraica  $(\mathbb{Z}_p/m(x), +, \cdot)$ , amb la suma i productes de polinomis modulars és un anell commutatiu amb unitat.

#### Exemple 2.14 Operacions a $(\mathbb{Z}_2/(x^3 + x + 1), +, \cdot)$

Com que estem a  $\mathbb{Z}_2$  i el polinomi és de grau 3 el conjunt tindrà un total de  $2^3$  elements que seran els següents:

$$\mathbb{Z}_2/(x^3 + x + 1) = \{0, 1, x, x + 1, x^2, x^2 + 1, x^2 + x, x^2 + x + 1\}$$

Donats els elements:

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

$$b(x) = x + 1$$

Suma de dos elements:

$$a(x) + b(x) = x^2 + x + 1 + x + 1 = x^2 + 2x + 2 = x^2, \text{ ja que estem a } \mathbb{Z}_2 \text{ i } 2 = 0 \pmod{2}.$$

Producte de dos elements:

$$a(x) \cdot b(x) = (x^2 + x + 1) \cdot (x + 1) = x^3 + 2x^2 + 2x + 1 = x^3 + 1 = x, \text{ on l'última igualtat s'obté de dividir el polinomi } x^3 + 1 \text{ entre } x^3 + x + 1 \text{ i quedar-se amb el reste, que val } x.$$

**Exercici 2.8** Quants elements té el conjunt  $\mathbb{Z}_2/(x^6 + x + 1)$ ?

**Exercici 2.9** Realitza els següents càlculs a  $\mathbb{Z}_3/(x^2 + x + 1)$

- $(x + 1) + (2x + 1)$
- $(x + 1) \cdot (x + 3)$
- $\frac{x+1}{2x+2}$

En el cas dels nombres enters hem vist que el concepte de nombre primer ens servia per definir estructures algebraiques a  $\mathbb{Z}_p$  que fossin cossos. Per tal de poder tenir la mateixa equivalència en els polinomis ens caldrà tenir un concepte similar al de nombre primer però per als polinomis. És el concepte de polinomi irreductible.

**Definició 2.15** Un polinomi  $a(x) \in \mathbb{Z}_p[x]$  és **irreductible** a  $\mathbb{Z}_p$  si al descomposar-lo com  $a(x) = b(x) \cdot c(x)$  amb  $b(x), c(x) \in \mathbb{Z}_p[x]$  aleshores  $b(x)$  o  $c(x)$  són constants, és a dir  $b(x) \in \mathbb{Z}_p$  o  $c(x) \in \mathbb{Z}_p$ .

Amb aquesta definició ja estem en condició de poder definir cossos finits de mida  $p^k$ .

**Teorema 2.10** Donat un nombre  $p$  primer i un polinomi  $m(x) \in \mathbb{Z}_p$  irreductible a  $\mathbb{Z}_p$  i de grau  $k$ , aleshores l'estructura algebraica  $(\mathbb{Z}_p/m(x), +, \cdot)$ , amb la suma i productes de polinomis modulars és un cos finit amb  $p^k$  elements.

Fixem-nos que si l'estructura algebraica  $(\mathbb{Z}_p/m(x), +, \cdot)$  és un cos, podem calcular l'invers de qualsevol dels seus elements. Per fer-ho, simplement utilitzarem les mateixes tècniques que hem descrit per als enters, però en aquest cas operant amb polinomis.

### Exemple 2.15 Càlcul d'inversos amb polinomis

Suposem l'estructura algebraica  $(\mathbb{Z}_2/(x^3 + x + 1), +, \cdot)$ . Com que  $x^3 + x + 1$  és irreductible a  $\mathbb{Z}_2$ , aquesta estructura algebraica és un cos. Calculem l'invers de l'element  $a(x) = x^2 + x + 1$ , és a dir, hem de trobar el polinomi  $b(x) \in \mathbb{Z}_2/(x^3 + x + 1)$  tal que  $a(x) \cdot b(x) = 1$ . Per fer-ho, hem de calcular els elements de la identitat de Bezout entre  $a(x)$  i  $X^3 + x + 1$  que és el mòdul on estem treballant.

Si calculem les divisions successives de l'algorisme d'Euclides per trobar el màxim comú divisor obtenim:

$$x^3 + x + 1 = (x^2 + x + 1) \cdot (x + 1) + x$$

$$x^2 + x + 1 = (x) \cdot (x + 1) + 1$$

tenim que el  $\gcd(x^3 + x + 1, x^2 + x + 1) = 1$ , com ja sabíem. Ara si aïllem els dos residus de cada equació:

$$1 = x^2 + x + 1 - (x) \cdot (x + 1)$$

$$x = x^3 + x + 1 - (x^2 + x + 1) \cdot (x + 1)$$

A partir d'aquestes equacions, podem calcular els coeficients de la identitat de Bezout, igual que hem fet en els enters. Fixeu-vos, que com que els coeficients del polinomi són elements de  $\mathbb{Z}_2$  no tenim en compte el signe ja que  $-1 = 1 \pmod{2}$ . Així obtenim que:

$$1 = (x^3 + x + 1) \cdot (x + 1) + (x^2 + x + 1) \cdot (x^2)$$

Si prenem aquesta equació mòdul  $x^3 + x + 1$  tindrem que el primer terme val 0 i per tant ens queda:

$$1 = (x^2 + x + 1) \cdot (x^2)$$

és a dir, l'invers de l'element  $x^2 + x + 1$  que buscàvem és el polinomi  $x^2$ . Ho podem comprovar fent el producte de  $x^2 + x + 1$  per  $x^2$  i veient que si el dividim pel mòdul,  $x^3 + x + 1$ , el reste ens dona 1.

## 2.2 Nombres primers

Els nombres primers han estat estudiats abastament ja que són la base de tots els nombres, donat que qualsevol nombre enter es pot descompondre de manera única com a producte de primers. Tot i això, hi ha moltes propietats d'aquest tipus de nombres que encara no s'han pogut sistematitzar i grans qüestions obertes de la matemàtica giren al voltant dels nombres primers.

Com ja hem indicat anteriorment en la Definició 2.9, un **nombre primer** és aquell enter positiu que només es pot dividir per ell mateix i per la unitat.

### Primers de Mersenne

Els primers del tipus  $2^p - 1$  s'anomenen primers de Mersenne. Aquests tipus de nombres són primers només en el cas que  $p$  sigui primer, però no es cert que qualsevol valor  $p$  primer generi un nombre  $2^p - 1$  primer. Per exemple, per  $p = 2$  sí que es compleix ja que  $2^2 - 1 = 3$  que és primer. Però per  $p = 11$  no és cert, perquè  $2^{11} - 1 = 2047$  que no és primer.

Els nombres primers s'estudien des de l'antiguitat i ja Euclid, 300 anys a.C. va demostrar que hi havia infinits nombres primers. Tot i això, i malgrat els diferents estudis sobre nombres primers, encara no s'ha pogut establir una fórmula que permeti donar la seqüència de nombres primers, i per tant, la única forma de trobar-los és anar generant nombres i comprovar si són primers o no ho són. En aquest sentit, el primer més gran que s'ha trobat (al gener del 2016) és el nombre  $2^{74,207,281} - 1$  que és un primer de Mersenne de més de 22 milions de xifres.

Tot i que no es coneix quina és la seqüència de nombres primers, sí que hi ha alguns resultats que permeten tenir estimacions sobre el nombre de primers que hi ha en un interval. Per exemple, el teorema dels nombres primers ens en dona una aproximació.

**Teorema 2.11 — Teorema dels nombres primers.** Sigui  $\pi(n)$  el nombre de primers més petits que un valor  $n$ , aleshores, es compleix que:

$$\lim_{n \rightarrow \infty} \frac{\pi(n)}{\frac{n}{\ln(n)}} = 1$$

És a dir, podem aproximar el nombre de primers més petits que  $n$  calculant-ne el seu logaritme. En la següent taula es pot veure la diferència entre aquesta aproximació i el nombre real de primers que existeixen per a valors petits on s'han pogut calcular tots els nombres primers i comptar-los.

$n$	$\pi(n)$	$\frac{n}{\ln(n)}$
10	4	4.34
100	25	21.7
1000	168	144.8
$10^6$	78498	72382
$10^9$	50847478	48254942

Si ens fixem amb la taula, tot i que la diferència entre l'aproximació i el nombre real de primers sembla que cada vegada sigui més distant, si calculem el quocient dels dos valors veurem que efectivament cada vegada és més proper a 1.

A l'hora de trobar nombres primers, el teorema que acabem d'enunciar ens dóna una estimació de la probabilitat que, donat un nombre aleatori qualsevol, aquest sigui primer. Efectivament, utilitzant l'aproximació que ens proporciona aquest teorema, sabem que trobar un nombre primer entre  $n$  i  $2n$  té una probabilitat de  $p = \frac{2}{\ln(2n)} - \frac{1}{\ln(n)}$  ja que el nombre total de primers en l'interval serà  $\frac{2n}{\ln(2n)} - \frac{n}{\ln(n)}$  i això caldrà dividir-ho pel nombre d'elements de l'interval, és a dir  $n$ . Per posar aquests valors en context, si fem els càlculs, veurem que la probabilitat que donat un valor triat aleatòriament entre  $1 \cdot 10^{20}$  i  $2 \cdot 10^{20}$  sigui primer és d'un 2%. Per tant, si el procés per aconseguir-ne un passa per seleccionar-ne un a l'atzar, mirar si és primer i sinó buscar-ne un altre, clarament el cost de mirar si un nombre és primer, cal que sigui computacionalment reduït si volem ser eficients en la generació de nombres primers.

### 2.2.1 Tests de primalitat

Com veurem al llarg del llibre, per a diferents criptosistemes i protocols criptogràfics, és necessari poder disposar de nombres primers molt grans. A l'hora de generar-los, donat que no tenim una fórmula que ens en doni la seqüència, el procés que es realitza consisteix a seleccionar un nombre aleatori molt gran i verificar-ne si és primer o no.

Per tal de comprovar si un nombre és primer s'utilitzen els test de primalitat, que no són més que algorismes que reben com a entrada un nombre i proporcionen com a sortida informació sobre la condició de primer del nombre donat. De test de primalitat n'hi ha de dos tipus: deterministes i probabilístics.

**Un test de primalitat determinista** és aquell que donat un nombre natural ens indica, de manera inequívoca, si és o no un nombre primer.

Una manera de determinar la primalitat d'un nombre seria fent-ne la descomposició en nombres primers. Si aquesta descomposició retorna més d'un factor diferent al propi nombre la conclusió és que el nombre no és primer i, en cas que els factors retornats siguin el propi nombre i l'1 es determinarà que el nombre sí que és primer.

Per exemple, podem utilitzar l'algorisme de factorització per prova de divisions, com el que es mostra en el següent codi en SAGE<sup>5</sup>, que ens retornarà un sol valor (el nombre proporcionat) en

<sup>5</sup>El SAGE és un programari matemàtic molt potent de codi obert basat en Python. Podeu descarregar-vos-el de <http://www.sagemath.org/>. Tots els fragments de codi que es mostren en aquest capítol són codificats en SAGE.

cas que aquest sigui primer.

```
def trial_division(n):
    if n < 2:
        return []
    prime_factors = []
    for p in primes_first_n(int(n**0.5)):
        if p*p > n: break
        while n % p == 0:
            prime_factors.append(p)
            n //= p
    if n > 1:
        prime_factors.append(n)
    return prime_factors
```

Evidentment aquest algorisme no és gens eficient, i per tant, per a verificar la primalitat de nombres molt grans és totalment desaconsellable. En l'actualitat, l'algorisme més eficient que proporciona un test de primalitat determinista és el proposat pels matemàtics indis Agrawal, Kayal i Saxena l'any 2002. Malgrat ser el test determinista més eficient, no s'utilitza a la pràctica ja que per a valors elevats els temps de resposta són massa grans.

Per tal d'obtenir test de primalitat amb una complexitat suficientment baixa per a poder generar nombres primers prou grans en un interval de temps prou petit cal recórrer als test de primalitat probabilístics.

Un **test de primalitat probabilístic** és aquell que donat un nombre natural ens indica si és primer amb una certa probabilitat.

Així doncs, un test de primalitat probabilístic ens pot indicar que un nombre és primer sense que realment ho sigui. De fet un test de primalitat probabilístic ens retornara el que es coneix com a pseudo-primer. El gran avantatge dels tests probabilístics és que són extremadament eficients i, a més, es pot determinar el valor de la probabilitat amb el que es poden equivocar i fer-lo tant petit es vulgui.

### Test de primalitat de Fermat

El test de primalitat de Fermat és un test de primalitat probabilístic basat en el teorema petit de Fermat. El teorema petit de Fermat és un cas particular del Teorema d'Euler que hem vist en apartats anteriors.

**Teorema 2.12 — Teorema petit de Fermat.** Sigui  $p$  un nombre primer, aleshores  $a^{p-1} = 1 \pmod{p}$  per a qualsevol valor  $a$  tal que  $1 \leq a < p$ .

En base a aquest teorema, podem definir el test de primalitat de Fermat de la següent manera:

```
def Fermat_test(n,k):
    if n <= 3:
        return str(n)+' es primer '
    for i in range(k):
        a = randint(2,n-2)
        if (a^(n-1))%n != 1:
            return str(n)+' es compost '
    p = numerical_approx(1-(1/2)^k)
    return str(n)+' es primer amb probabilitat '+ str(p)
```

#### Nombres de Carmichael

Els nombres de Carmichael són nombres extremadament rars. Un nombre de Carmichael  $n$ , tot i no ser primer, compleix la congruència de Fermat per a tots els valors  $a$ , tals que  $1 \leq a < n$  i  $\text{mcd}(n,a) = 1$ . Per aquest motiu, el test de primalitat Fermat aplicat al nombre de Carmichael  $n = 340561$  (que per tant és compost) ens pot arribar a donar que és un nombre primer amb probabilitat 0,999.

Fixeu-vos que la idea és prendre el nombre que volem analitzar i assignar-lo com al mòdul de l'equació. Pel teorema petit de Fermat sabem que si el nombre és primer, l'equació modular sempre ens donarà 1. Ara bé, si el nombre no és primer l'equació modular pot donar 1 o pot donar un valor diferent de 1. A més, si  $n$  no és primer, en general, la meitat dels valors  $a$  més petits que  $n$  complirà l'equació i l'altra meitat no. Això ens porta a assegurar que si anem triant valors  $a$  diferents la probabilitat que l'equació sigui certa sense que  $n$  sigui primer és cada vegada més petita. En particular, la probabilitat es redueix a la meitat. Per aquest motiu, si repetim el test de Fermat  $k$  vegades i ens indica que el valor  $n$  és primer, la probabilitat que aquest ho sigui serà  $1 - (\frac{1}{2})^k$ .

#### Test de primalitat de Miller-Rabin

El test de primalitat de Miller-Rabin és un test que combina la condició del teorema petit de Fermat amb la particularitat dels residus quadràtics en aritmètica modular. Tal i com hem vist en l'Apartat 2.1.3, en el cas que  $n$  és primer l'equació  $x^2 = 1 \pmod{n}$  té únicament dues solucions, mentre que si  $n$  és compost en té quatre. Així, aquest fet, juntament amb el teorema petit de Fermat es poden unir creant el test de primalitat de Miller-Rabin implementat en el següent algorisme:

```
def Miller_Rabin_test(n,k):
    tmp = n-1
    s = 0
    while tmp%2 == 0:
        tmp = tmp // 2
        s = s + 1
    r = (n-1) / (2^s)
    for i in range(k):
        a = randint(2,n-2)
        y = a^r%n
        if (y != 1) and (y != n-1):
            j = 1
            while (j >= (s-1)) and (y != (n-1)):
                y = (y^2)%n
                if y==1:
                    return str(n)+'_es_compost '
                j = j+1
            if y != n-1:
                return str(n)+'_es_compost '
    p = numerical_approx(1-(1/4)^k)
    return str(n)+'_es_primer_amb_probabilitat_'+ str(p)
```

L'avantatge d'aquest test respecte el de Fermat és que no està afectat pels nombres de Carmichael. A més, a cada iteració la probabilitat d'error disminueix en  $1/4$  en comptes d' $1/2$  aconseguint una probabilitat més alta d'encertar un primer amb menys iteracions que el test de Fermat. En l'actualitat, per la seva eficiència, aquest test, o alguna de les seves variants, és un dels més utilitzats en les aplicacions criptogràfiques.

## 2.3 Problemes matemàtics difícils

Com veurem al llarg d'aquest llibre, la seguretat dels algoritmes criptogràfics que es fan servir avui en dia recau en la dificultat que un atacant pugui realitzar els càlculs necessaris per trencar el criptosistema. Així, tal i com hem definit anteriorment, la seguretat de la majoria dels criptosistemes moderns és una seguretat computacional i no pas una seguretat teòrica.

Per tal de poder definir problemes que siguin difícils de resoldre per un atacant, ens caldrà primer definir què vol dir que un problema sigui difícil des d'un punt de vista computacional. A continuació, repassarem diferents funcions matemàtiques que presenten certa unidireccionalitat en el sentit que el seu càlcul en una direcció és molt simple però el càlcul de la seva inversa és molt complicat, fet que s'utilitza en el disseny de criptosistemes i protocols criptogràfics.

### 2.3.1 Complexitat d'un algorisme

La teoria de la complexitat algorísmica és molt complexa en si mateixa, de manera que en aquest apartat només en donarem unes nocions molt bàsiques.

La complexitat de càlcul d'un algorisme es mesura pel temps  $T$  que requereix la seva execució i s'expressa com a funció de la mida  $n$  de l'entrada de l'algorisme. Més que fer servir complexitats exactes que s'expressarien com a  $f(n)$ , se solen emprar ordres de magnitud, és a dir  $\mathcal{O}(g(n))$ , de tal manera que  $f(n) = \mathcal{O}(g(n))$  vol dir que hi ha constants  $c$  i  $n_0$  tals que:

$$f(n) \leq c|g(n)| \text{ ; per a } n \geq n_0$$

El propòsit que hi ha en fer servir ordres de magnitud és que l'explicitació de  $g(n)$  sigui més simple que  $f(n)$ .

#### Exemple 2.16 Ordre de la complexitat

Si la complexitat exacta d'un algorisme és  $f(n) = 36n + 10$ , podem escriure que  $f(n) = \mathcal{O}(n)$ , ja que

$$36n + 10 \leq 37n \text{ ; per a } n \geq 10$$

Per la mateixa raó, si  $f(n)$  és un polinomi de grau  $t$  en  $n$ , podem escriure  $f(n) = \mathcal{O}(n^t)$ .



Un **algorisme polinòmic** és aquell que el seu temps d'execució és  $T = \mathcal{O}(n^t)$  per a alguna constant  $t$ . En el cas que  $t = 0$  direm que l'algorisme és constant, lineal si  $t = 1$ , quadràtic si  $t = 2$ , etc.

Un **algorisme exponencial** és aquell que el seu temps d'execució és  $T = \mathcal{O}(t^{h(n)})$  per a alguna constant  $t$  i un polinomi  $h(n)$ .

Per a valors d' $n$  grans, les diferents classes de complexitat impliquen temps molts diferents d'execució. Per exemple, si suposem una màquina capaç d'executar  $10^{12}$  instruccions per segon, la taula següent ens mostra els temps d'execució per a les classes d'algorismes que acabem de definir.

Classe	Complexitat	Operacions per $n = 10^{12}$	Temps
Constant	$\mathcal{O}(1)$	1	$10^{-12}$ segons
Lineal	$\mathcal{O}(n)$	$10^{12}$	1 segon
Quadràtic	$\mathcal{O}(n^2)$	$10^{24}$	31.709 anys
Cúbic	$\mathcal{O}(n^3)$	$10^{36}$	$3,17 \cdot 10^{16}$ anys
Exponencial	$\mathcal{O}(2^n)$	$10^{3 \cdot 10^{11}}$	$10^{2,999 \cdot 10^{11}}$ milions d'anys.

### 2.3.2 Producte de primers i factorització d'enters

Un dels problemes matemàtics difícil és la factorització d'enters. Si tenim dos nombres primers  $p$  i  $q$  és molt fàcil calcular el seu producte  $n = p \cdot q$ . Això és així independentment de la mida dels valors  $p$  i  $q$  perquè els algorismes que realitzen la multiplicació d'enters són algorismes molt eficients i per tant, la mida dels nombres afecta molt poc al temps de resolució del producte.

Ara bé, donat un valor  $n$  que sabem que és producte de dos primers, és molt difícil trobar quins són aquests dos primers, és a dir, factoritzar-los, en el cas que  $n$  sigui prou gran.

Podeu veure la diferència entre aquestes dues operacions executant les següents comandes de Sage.

```
p=next_prime(2^100)
q=next_prime(2^101)
print 'Temps per multiplicar ', p, ' per ', q, ':'
time n=p*q
print 'Temps per factoritzar ', n, ':'
time factor(n)
```

D'algorismes per a la factorització d'enters n'hi ha de diferents tipus i la seva complexitat depèn de la forma dels primers que generen el valor a factoritzar. Si ens centrem en algorismes de factorització genèrics, per a valors de  $p$  i  $q$  propers i grans però sense cap caracterització concreta, el millor algorisme de factorització que es coneix s'anomena sedàs de cos de nombre general, en anglès *general number field sieve (GNFS)* i té una complexitat  $\mathcal{O}(e^{(\sqrt[3]{\frac{64}{9}} + \mathcal{O}(1))(\ln(n)^{\frac{1}{3}}(\ln(\ln(n)))^{\frac{2}{3}})})$

Com veurem en els propers capítols, la dificultat en la factorització d'enters de mida molt gran és la base de la seguretat del criptosistema RSA.

### 2.3.3 Exponenciació i logaritme discret

Els enters modulars,  $\mathbb{Z}_p$  amb  $p$  primer, són un grup multiplicatiu cíclic. Això vol dir que podem trobar un element  $g$  que s'anomena generador, tal que les seves potències generen tots els elements de  $\mathbb{Z}_p$ , llevat del zero. Per exemple, si considerem  $\mathbb{Z}_7 = \{0, 1, 2, 3, 4, 5, 6\}$  veiem que el 3 és un generador perquè  $\mathbb{Z}_7 = \{0, 1 = 3^6, 2 = 3^2, 3 = 3^1, 4 = 3^4, 5 = 3^5, 6 = 3^3\}$ . A més, la distribució de les potències del generador entre els elements de  $\mathbb{Z}_p$  és una distribució uniforme.

En aquest entorn és on trobem un altre dels problemes matemàtics que s'utilitzen en criptografia. De nou, tenim una operació molt fàcil de realitzar: donats dos elements  $x, y \in \mathbb{Z}_p$  calcular la seva potència, és a dir  $z = x^y \pmod{p}$ . Aquesta operació és molt eficient de realitzar perquè l'únic càlcul que realitzem són productes, amb una divisió per reduir el resultat al mòdul desitjat. Ara bé, l'operació inversa, la que donats dos elements  $x, z \in \mathbb{Z}_p$  permet trobar l'element  $y$  tal que  $z = x^y \pmod{p}$  és un problema pel qual no se'n coneix cap algorisme eficient. Fixeu-vos que aquest càlcul és l'equivalent a calcular el logaritme de  $z$  en base  $x$ , però en els enters mòdul  $p$ . Per això, aquest problema se'l coneix com a problema del logaritme discret.

El problema del logaritme discret és la base de la seguretat de diferents esquemes i protocols criptogràfics, els més coneguts dels quals són l'intercanvi de claus de Diffie i Hellman i el criptosistema d'ElGamal.

### 2.3.4 Quadrats i arrels quadrades modulars

Un altre problema matemàtic que s'utilitza en criptografia és el càlcul d'arrels quadrades en un anell multiplicatiu sobre  $\mathbb{Z}_n$  quan el valor  $n$  és un producte de dos primers  $p$  i  $q$ .

Com ja hem comentat anteriorment, multiplicar dos nombres és molt ràpid, de manera que calcular el quadrat d'un nombre ha de ser forçosament també molt ràpid, perquè és el producte d'un nombre per ell mateix. A més, si una vegada hem fet el producte en volem calcular el seu equivalent mòdul  $n$ , això també és molt ràpid, perquè només cal que dividim el resultat pel mòdul i ens quedem amb el residu. Per tant, calcular quadrats a  $\mathbb{Z}_n$  és pot fer de manera molt eficient. Ara bé, l'operació inversa, és a dir, donat un element  $x \in \mathbb{Z}_n$  trobar-ne l'arrel quadrada (l'element  $y$  tal que  $x = y^2 \pmod{n}$ ), és una operació molt costosa. De fet, tal i com s'enuncia en el Teorema 2.8, aquest problema és equivalent a factoritzar ja que per calcular les arrels d'un valor a  $\mathbb{Z}_n$ , on  $n = pq$ , ja hem vist que ens caldrà trobar les arrels quadrades a  $\mathbb{Z}_p$  i a  $\mathbb{Z}_q$ , i per tant, ens cal factoritzar  $n$ .

## 2.4 Resum

En aquest capítol hem presentat els conceptes matemàtics bàsics utilitzats en criptografia, centrats en l'aritmètica modular. Conceptes com la divisibilitat de nombres enters, el màxim comú divisor o el teorema d'Euler són cabdals per poder comprendre les operacions criptogràfiques que realitzen els algorismes de xifrat. Així, l'aritmètica modular és la base que ens permetrà entendre el funcionament de la majoria dels criptosistemes utilitzats en l'actualitat, tant els de criptografia de clau simètrica, com l'AES, com els de clau pública com l'RSA o ElGamal.

Ja centrats en la criptografia de clau pública, és important tenir clares les característiques dels nombres primers, ja que aquests acostumen a ser la matèria prima en la que es basen els criptosistemes. Aquests valors sovint formen part de les claus o dels paràmetres dels esquemes i per tant conèixer-ne la seva distribució i comprendre com es poden obtenir és vital per poder implementar qualsevol criptosistema de clau pública.

Finalment, i continuant amb la criptografia de clau pública, hem analitzat diferents problemes matemàtics que tenen una asimetria en la seva resolució i que es fan servir en criptosistemes de clau pública. Com hem vist, aquestes problemes presenten una simplicitat d'execució quan els mirem en un sentit però són d'una dificultat extrema quan els volem realitzar en el sentit invers. Per exemple, multiplicar dos primers grans  $p$  i  $q$  pot ser immediat mentre que trobar-ne els que componen un nombre  $n$  pot implicar càlculs de milers d'anys. És important conèixer quins són aquests problemes matemàtics per tal d'entendre quines són les bases de seguretat dels criptosistemes que els fan servir.

## 2.5 Solucions dels exercicis

### Exercici 2.1:

$\text{mcd}(35, 48) = 1$  ja que si calculem les divisions successives tenim:  $48 = 35 \cdot 1 + 13$

$$35 = 13 \cdot 2 + 9$$

$$13 = 9 \cdot 1 + 4$$

$$9 = 4 \cdot 2 + 1$$

$$4 = 4 \cdot 1 + 0$$

i l'últim reste no nul és el 1.

### Exercici 2.2:

Per calcular el coeficients de la identitat de Bezout utilitzarem les igualtats de l'algorisme d'Euclides de l'exercici anterior i n'aïllarem el residu:

$$48 - 35 = 13$$

$$35 - (13 \cdot 2) = 9$$

$$13 - 9 = 4$$

$$9 - (4 \cdot 2) = 1$$

i substituïrem en cada equació el valor corresponent per acabar obtenint-ne una sola amb els valors 35 i 48:

$$1 = 9 - (4 \cdot 2) = 9 - ((13 - 9) \cdot 2) = (3 \cdot 9) - (13 \cdot 2) = (3(35 - (13 \cdot 2)) - (13 \cdot 2) = (35 \cdot 3) - (8 \cdot 13) = (3 \cdot 35) - (8(48 - 35))) = (11 \cdot 35) - (8 \cdot 48)$$

Així, tenim que

$$1 = 11 \cdot 35 + (-8) \cdot 48$$

i per tant els coeficients de la identitat de Bézout per a 35 i 48 són 11 i  $-8$  respectivament.

### Exercici 2.3:

$$\phi(527) = \phi(17 \cdot 31) = \phi(17) \cdot \phi(31) = (17 - 1) \cdot (31 - 1) = 16 \cdot 30 = 480.$$

### Exercici 2.4:

El conjunt  $\mathbb{Z}_{25}$  està format per tots els restes de dividir per 25, per tant tindrà 25 elements que són  $\mathbb{Z}_{25} = \{0, 1, 2, 3, 4, \dots, 22, 23, 24\}$

### Exercici 2.7:

L'estructura algebraica  $(\mathbb{Z}_{37}, +, \cdot)$  és un cos ja que el nombre 37 és primer. Això fa que tots els elements més petits que 37, és a dir tots els elements inclosos en  $\mathbb{Z}_{37}$ , siguin coprimers amb 37 i per tant tinguin invers, condició necessària i suficient perquè  $(\mathbb{Z}_{37}, +, \cdot)$  sigui un cos.

### Exercici 2.5:

L'invers de 7 a  $\mathbb{Z}_{37}$  val 16 ja que  $7 \cdot 16 = 112 = 1 \pmod{37}$ . El valor 16 el podem calcular de diferents maneres. Per exemple, calculant  $7^{\phi(37)-1} = 7^{35} = 16 \pmod{37}$  o bé calculant els coeficients de la identitat de Bezout de 7 i 37, és a dir  $7 \cdot 16 + 37 \cdot (-3)$ .

### Exercici 2.6:

Realitza els següents càlculs a  $\mathbb{Z}_{37}$

- $20 + 20 = 40 = 3 \pmod{37}$
- $20 \cdot 4 = 80 = 6 \pmod{37}$

- $20^2 = 400 = 30 \pmod{37}$
- $\frac{20}{7} = 20 \cdot 16 = 320 = 24 \pmod{37}$ , ja que 16 és l'invers de 7 tal i com hem calculat anteriorment.

**Exercici 2.8:**

El conjunt  $\mathbb{Z}_2/(x^6 + x + 1)$  té un total de  $2^6 = 64$  elements ja que són tots els polinomis de grau 5 amb coeficients binaris.

**Exercici 2.9:**

Realitza els següents càlculs a  $\mathbb{Z}_3/(x^2 + x + 1)$

- $(x + 1) + (2x + 1) = 2 \pmod{(x^2 + x + 1)}$
- $(x + 1) \cdot (x + 3) = x^2 + x = 2 \pmod{(x^2 + x + 1)}$
- $\frac{x+1}{2x+2} = (x + 1) \cdot x = 2 \pmod{(x^2 + x + 1)}$ , ja que  $x^2 + x + 1 = 0$  per ser el mòdul i per tant:  $x^2 + x = 2$  ja que els coeficients del polinomi són de  $\mathbb{Z}_3$ .

## 2.6 Bibliografia

**Shanks, D.** (1973). *Five number-theoretic algorithms*. In Proceedings of the Second Manitoba Conference on Numerical Mathematics (Winnipeg).