Hvad er matematik? 3
ISBN 9788770668781

Projekter: Kapitel 0. *Diskret matematik. Projekt 0.3 Galois-legemerne - et værktøj til fejlrettende QR-koder*

L&R Uddannelse
EGMONT

Projekt 0.3 Galois-legemerne $\mathbb{GF}\left[p^n\right]$ - et værktøj til fejlrettende QR-koder

Indhold

I dette projekt skal vi kigge på forskellige talsystemer med henblik på bedre at forstå fejlrettende koder. Men vi starter med de mest almindelige talsystemer, de hele tal, de rationale tal og de reelle tal, og arbejder os så frem mod de endelige talsystemer, de såkaldte Galois-legemer, der ligger bag de fejlrettende koder i fx de kvadratiske QR-koder.

Hvad er matematik? 3
ISBN 9788770668781

Projekter: Kapitel 0. *Diskret matematik. Projekt 0.3 Galois-legemerne - et værktøj til fejlrettende QR-koder*

L&R Uddannelse
EGMONT

1. De karakteristiske egenskaber ved de tre mest almindelige talsystemer $\mathbb{Z}$, $\mathbb{Q}$ og $\mathbb{R}$

De mest almindelige talsystemer er de hele tal $\mathbb{Z}$, de rationale tal $\mathbb{Q}$ og de reelle tal $\mathbb{R}$. De rummer alle tre en række egenskaber ved talsystemer, der er meget attraktive:

1.1 Den kommutative, associative og distributive lov for addition og multiplikation

I alle tre tilfælde bygger de på to regneoperationer + (plus/addition) og · (gange/multiplikation) med følgende egenskaber:

Regneoperationerne er *kommutative*, dvs. når vi lægger to tal sammen eller ganger to tal med hinanden er rækkefølgen af tallene ligegyldig:

$$a+b=b+a \qquad a\cdot b=b\cdot a$$

Regneoperationerne er *associative*, dvs. når vi lægger tre tal sammen (ved hjælp af to additioner) eller ganger tre tal med hinanden (ved hjælp af to multiplikationer), så er rækkefølgen af regneoperationerne ligegyldige, dvs.

$$a+b+c=(a+b)+c=a+(b+c) \qquad a\cdot b\cdot c=(a\cdot b)\cdot c=a\cdot(b\cdot c)$$

Regneoperationerne er *distributive*, dvs. når vi ganger ind i en sum sker det ledvis:

$$a\cdot(b+c)=a\cdot b+a\cdot c$$

Vi tænker sjældent over de to regneoperationer, men de er i virkeligheden meget fundamentale og forenkler fx ligningsløsning betydeligt.

Øvelse 1:
Indenfor de naturlige tal $\mathbb{N}$, har vi også en tredje regneoperation, potensopløftning $a\,\hat{}\,b=a^b$, hvor vi normalt foretrækker den sidste skrivemåde med løftet eksponent, men her også bruger den første med potenstegnet ^ for netop at fremhæve, at der er tale om en regneoperation.

a) Gør rede for, at potensopløftning *ikke* er kommutativ.

b) Gør rede for, at potensopløftning *ikke* er associativ.

c) Kan man i en vis forstand sige, at potensopløftning er distributiv med hensyn til multiplikation?

d) Kan man i en vis forstand sige, at potensopløftning er distributiv med hensyn til addition?

Det kan synes ret uskyldigt at potensopløftning på den måde er mere kompliceret end addition og multiplikation, men i den aksiomatiske opbygning af talteorien fører det til alvorlige vanskeligheder.

1.2 De modsatte regneoperationer: Subtraktion og division

Til hver af de to regneoperationer hører der nu en *modsat regneoperation*, subtraktion henholdsvis division. Til at begynde med lægger vi mærke til, at de gængse talsystemer, $\mathbb{Z}$, $\mathbb{Q}$ og $\mathbb{R}$, dels indeholder tallet 0 som er neutralt over for addition, dvs. der gælder

$$0+x=x \quad \text{for alle } x.$$

Dels indeholder de tallet 1, som er neutralt over for multiplikation, dvs.

$$1\cdot x=x \quad \text{for alle } x.$$

Med udgangspunkt i det neutrale element, kan vi nu indføre et inverst element. Ved addition hedder det inverse tal det modsatte tal $-x$, og det er karakteriseret ved

$$(-x)+x=0$$

Ved multiplikation hedder det modsatte tal det reciprokke tal x^{-1} og det er karakteriseret ved

$$x^{-1}\cdot x=1$$

Har vi først rådighed over et inverst element kan vi nu indføre den modsatte regneoperation, dvs. *subtraktion*, ved at lægge det modsatte tal til, dvs.

$$a - b \overset{def}{=} a + (-b)$$

Tilsvarende kan vi indføre division for alle tal forskellig fra 0 ved at gange det reciprokke tal på, dvs.

$$a / b = \frac{a}{b} \overset{def}{=} a \cdot b^{-1}$$

Tallet 0 har dog ikke noget reciprokt element, idet der gælder *nulreglen*:

$$0 \cdot x = x \text{ for alle } x.$$

Øvelse 2:

a) Gør rede for at *nulreglen* er en konsekvens af den distributive lov.

Indenfor talsystemet $\mathbb{Z}$, har alle tal x et modsat tal, $-x$, dvs. subtraktion er også veldefineret indenfor $\mathbb{Z}$. Men i almindelighed har et helt tal derimod *ikke* et reciprokt tal indenfor $\mathbb{Z}$. Man kan derfor ikke dividere indenfor de hele tal.

Indenfor talsystemerne $\mathbb{Q}$ og $\mathbb{R}$ har derimod alle tal x et modsat tal, $-x$, ligesom et hver tal x forskellig fra 0 har et reciprokt tal, x^{-1}. Såvel subtraktion som division er derfor veldefinerede indenfor de rationale tal og de reelle tal.

2. Ringe og legemer

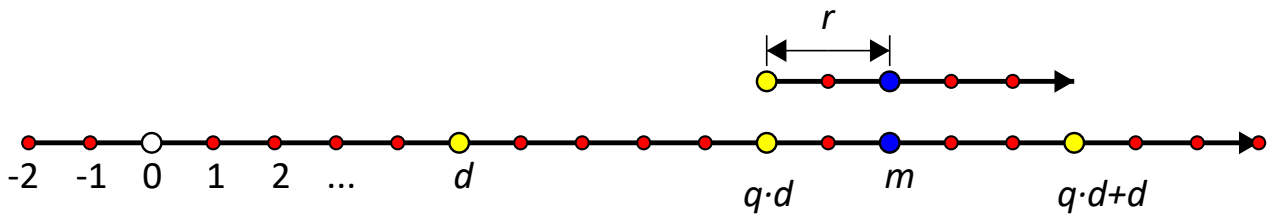
Definition 1: **Talringe**
Et talsystem med de to regneoperationer + og · kaldes en **talring**, hvis det indeholder de neutrale tal 0 og 1, og hvis ethvert tal x har et modsat tal $-x$, dvs. en talring er også lukket overfor subtraktion.

Definition 2: **Tallegemer**
Et talsystem med de to regneoperationer + og · kaldes et tallegeme, hvis det indeholder de neutrale tal 0 og 1, og hvis ikke blot har ethvert tal x et modsat tal $-x$, men ethvert tal x forskelligt fra 0 har også et reciprokt tal x^{-1}, dvs. et tallegeme er både lukket overfor subtraktion og division.

Det er da klart at de hele tal $\mathbb{Z}$ er et eksempel på en talring. Tilsvarende er de rationale tal $\mathbb{Q}$ og de reelle tal $\mathbb{R}$ eksempler på tallegemer. Men der findes mange flere eksempler! I det følgende styre vi mod at konstruere alle *endelige* tallegemer, men først skal vi se lidt nærmere på to fundamentale konstruktionsmetoder til at omdanne talringe til tallegemer!

2.1. Fra ring til legeme: Regning med restklasser

I den første konstruktionsmetode benytter vi os af en *divisionsalgoritme*. Som udgangspunkt tager vi heltalsringen $\mathbb{Z}$. Den er *ikke* lukket over division, men der findes ikke desto mindre en simpel *divisionsalgoritme*, som løst sagt fortæller, hvor mange gange q (kvotienten) et givet naturligt tal d (divisoren) går op i et andet givet helt tal m (dividenden), og hvilke *rest* r der så bliver til overs.



Ideen er at vi kigger på *multipla* af divisoren, dvs. d-tabellen eller hele tal på formen $q \cdot d$. De ligger ækvidistant på talaksen, idet afstanden mellem to successive multipla netop er d. Hvis vi lukker intervallet mellem to successive multipla i det nederste multiplum, indeholder intervallet altså netop d tal på formen

$$\{q \cdot d, q \cdot d + 1, q \cdot d + 2, \ldots, q \cdot d + (d-1)\}$$

Tallet *m* ligger da i netop et af disse intervaller, dvs. det kan netop skrives entydigt på formen

$$m = q \cdot d + r, \quad 0 \le r < d$$

Hvis resten er 0, siger vi at divisionen går op og tallet *d* kaldes en divisor. Tallet 1 går op i alle andre tal, men hvis divisoren *d* er forskellig fra 1 og *m* kaldes divisoren en *ægte divisor*. Vi siger da at tallet *m* er sammensat, dvs. det kan skrives som et produkt af to *mindre tal*.

2.1.1 Modulus-funktionen

Der findes en speciel funktion kaldet *modulus*, som udregner heltalsresten ved division. Typisk ser den således ud

$$\mathrm{mod}(m,d) = r$$

Vi finder fx

$$mod\left(45,7\right) = 3 \ ,$$

idet der jo gælder

$$45 = 42 + 3 = 6 \cdot 7 + 3$$

I abstrakt matematik bruger man ofte notationen *m* mod *d* i stedet for mod(m,d)!

Heltalsresten kan også findes ved *Euklids algoritme*, idet vi successivt trækker 7 fra indtil vi kommer under divisoren 7:

$$\left.\begin{aligned} 45-7&=38\\ 38-7&=31\\ 31-7&=24\\ 24-7&=17\\ 17-7&=10\\ 10-7&=3 \end{aligned}\right\} \text{6 gange}$$

Vi ser da at vi kan trække 7 fra 6 gange, dvs. 7 går 6 gange op i 45, og at resten bliver 3.

Det er ikke alle talringe, der understøtter Euklids algoritme, der jo bygger på at indenfor de hele tals ring $\mathbb{Z}$ bliver der mindre og mindre tilovers, når vi successivt trækker divisoren fra. Der skal altså være en veldefineret ordningsstruktur, der tillades os at tale om at et tal kan være mindre end eller større end et andet tal.

Det er altså *ikke* alle talringe, der har en divisionsalgoritme, så her udnytter vi nogle helt særlige forhold ved heltalsringen $\mathbb{Z}$!

En anden speciel egenskab ved heltalsringen $\mathbb{Z}$, er at der ikke findes nogen *nuldivisorer*, dvs. der findes ingen naturlige tal, der går op i 0. Hvis et produkt $a \cdot b$ giver 0 må mindst en af faktorerne være nul. Heller ikke dette kan man forvente skal gælde i almene talringe, hvilket vi skal se eksempler på lige om lidt!

Definition 3: Restklasserne modulo *n*
Lad nu *n* være et naturligt tal større end 1. Vi kigger da på resterne ved division med *n*. Der er netop *n* sådanne rester og de udgør talsystemet $\mathbb{Z}_n$ (også kaldet restklasserne modulo *n*).

Vi stiler nu mod at vise, at restklasserne modulo det naturlige tal *n*, dvs. talsystemet $\mathbb{Z}_n$, (hvor *n* er større end 1) udgør en talring.

2.1.2 Addition og multiplikation af restklasser

Vi skal først vise, at vi kan lægge to restklasser sammen og at vi kan gange to restklasser med hinanden.

Det sker ved at bemærke at hvis vi lægger to hele tal sammen afhænger resten kun af restklasserne for de to hele tal, dvs.

Sætning 2: Sum af restklasser
$(a+b) \bmod n = (a \bmod n) + (b \bmod n) \mod n$

Bevis: Hvis *a* har resten r_a modulo *n* og b har resten r_b modulo *n*, gælder der ifølge divisionsalgoritmen

$$\left.\begin{aligned} a &= q_a \cdot n + r_a \\ b &= q_b \cdot n + r_b \end{aligned}\right\} \quad a+b = (q_a \cdot n + r_a) + (q_b \cdot n + r_b) = (q_a + q_b) \cdot n + (r_a + r_b)$$

Det viser netop at *n* går $q_a + q_b$ gange op i $a+b$ med resten $r_a + r_b$, men resten behøver ikke være mindre end *n*, selv om de to individuelle rester nødvendigvis er mindre end *n*. Vi skal derfor eventuelt trække *n* fra endnu engang.

Sætning 3: Produkt af restklasser
$(a+b) \bmod n = (a \bmod n) + (b \bmod n) \mod n$

Bevis: Hvis *a* har resten r_a modulo *n* og b har resten r_b modulo *n*, gælder der ifølge divisionsalgoritmen

$$\left.\begin{aligned} a &= q_a \cdot n + r_a \\ b &= q_b \cdot n + r_b \end{aligned}\right\} \quad a \cdot b = (q_a \cdot n + r_a) \cdot (q_b \cdot n + r_b) = (q_a \cdot q_b \cdot n + q_a \cdot r_b + q_b \cdot r_a) \cdot n + (r_a \cdot r_b)$$

Det viser netop at *n* går $q_a \cdot q_b \cdot n + q_b \cdot r_a + q_b \cdot r_b$ gange op i $a \cdot b$ med resten $r_a \cdot r_b$, men resten behøver ikke være mindre end *n*, selv om de to individuelle rester nødvendigvis er mindre end *n*. Vi skal derfor eventuelt trække *n* fra endnu nogle gange for at finde heltalsresten. Det kræver den ekstra modulo-udregning til sidst!

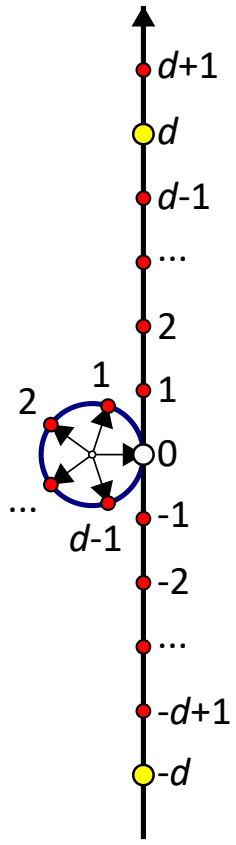
Addition af to restklasser er i en vis forstand triviel. Man kan opfatte restklassemængden $\mathbb{Z}_n$ som en talcirkel, idet man tager den sædvanlige tallinje og snor den op på en cirkel med omkredsen *n*, så tallet *n* netop falder i tallet 0. Alle tallene på tallinjen falder da netop i deres tilhørende restklasse på talcirklen, som netop rummer *n* gitterpunkter $0,1,2,\ldots,n-1$.

Addition med restklassen *r* svarer da netop til en drejning i positiv retning med den drejningsvinkel, der fører restklassen 0 over i restklassen *r*.

Subtraktion med restklassen *r* svarer tilsvarende til en drejning i negativ retning med den drejningsvinkel, der fører restklassen *r* over i restklassen 0.

Hvor addition og subtraktion på den almindelige tallinje svarer til parallelforskydninger svarer addition og subtraktion på talcirklen altså til *drejninger*.

Helt så simpelt går det desværre ikke med at tolke multiplikation på talcirklen. Godt nok kan man forestille sig at man strækker en cirkel ud fra et begyndelsespunkt på cirklen. Men det er ikke helt så oplagt at gitterpunkterne netop falder på gitterpunkter



Vi kan nu nemt konstruere additionstabeller og multiplikationstabeller for restklasser. Vi viser princippet bar restklasser modulo 7.

I regnearket afsætter vi tabeller med restklasserne $\{0,1,2,3,4,5,6\}$:

	A	B	C	D	E	F	G	H	I	J
1	Addition									
2		7	0	1	2	3	4	5	6	
3		0								
4		1								
5		2								
6		3								
7		4								
8		5								
9		6								
10	Multiplikation									
11		7	0	1	2	3	4	5	6	
12		0								
13		1								
14		2								
15		3								
16		4								
17		5								
18		6								
19										

For at konstruere en additionstabel skal vi så blot vise formlen i det øverste venstre hjørne $C3$ af tabellen:

$$C3 = \mathrm{mod}(\$B3 + C\$2, \$B\$2)$$

Her skal dollartegnene sikre at vi hele tiden lægger tal sammen, der stammer fra søjle *B* og række 2. Tilsvarende skal vi hele tiden regne modulo 7, dvs. det sidste tal skal låses til cellen *B*2!

Denne formel trækkes da første nedad i tabellen og derefter på tværs. Det samme gøres med multiplikationstabellen:

	A	B	C	D	E	F	G	H	I	J
1	Addition									
2		7	0	1	2	3	4	5	6	
3		0	0	1	2	3	4	5	6	
4		1	1	2	3	4	5	6	0	
5		2	2	3	4	5	6	0	1	
6		3	3	4	5	6	0	1	2	
7		4	4	5	6	0	1	2	3	
8		5	5	6	0	1	2	3	4	
9		6	6	0	1	2	3	4	5	
10	Multiplikation									
11		7	0	1	2	3	4	5	6	
12		0	0	0	0	0	0	0	0	
13		1	0	1	2	3	4	5	6	
14		2	0	2	4	6	1	3	5	
15		3	0	3	6	2	5	1	4	
16		4	0	4	1	5	2	6	3	
17		5	0	5	3	1	6	4	2	
18		6	0	6	5	4	3	2	1	
19										

Øvelse 3: $\mathbb{Z}_7$

a) Kig lidt på tabellen og prøv at regne nogle af cellerne ud i hovedet! Hvorfor gælder der fx 3·4 = 5, når vi regner modulo 7?

b) Hvordan kan man se af tabellerne at addition og multiplikation er kommutative?

c) Hvordan kan man se at 0 er et neutralt element overfor addition og at 1 er et neutralt element over multiplikation?

d) Har ethvert tal et modsat tal (dvs. inverst element overfor addition)?

e) Har ethvert tal forskelligt fra 0 et reciprokt tal (dvs. et inverst element overfor multiplikation)?

f) Er $\mathbb{Z}_7$ en talring? Er det et tallegeme?

Øvelse 4: $\mathbb{Z}_6$

a) Konstruér nu tilsvarende tabeller modulo 6.

b) Besvar de samme spørgsmål som ovenfor for restklassemængden $\mathbb{Z}_6$.

2.2 Restklasseringen $\mathbb{Z}_n$, hvor *n* er et naturligt tal

På basis af sådanne øvelser skulle det nu ikke komme som en overraskelse at der gælder følgende sætning:

Sætning 4:
Restklassemængden $\mathbb{Z}_n$ er en talring.

Bevis (skitse):
Vi har allerede indført to regneoperationer på restklasser, som arver de nødvendige egenskaber (kommutativitet, associativitet og distributivitet) fra de sædvanlige regneregler.

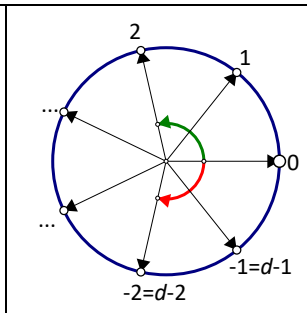
For at vise at $\mathbb{Z}_n$ er en talring, skal vi derfor blot vise at enhver restklasse *r* har en modsat restklasse overfor addition, dvs. en restklasse *s*, hvor der gælder

$$r + s = 0 \bmod n .$$

Men der gælder oplagt

$$s = n - r$$

På talcirklen ligger de to modsatte restklasser lige overfor hinanden ved spejling i hoveddiameteren gennem 0. De tilhørende drejninger foregår da med lige store og modsatrettede drejningsvinkler, dvs. de ophæver netop hinanden.



2.3 Restklasselegemet $\mathbb{Z}_p$, hvor *p* er et primtal

Mere overraskende er nok den følgende sætning:

Sætning 5:
Restklassemængden $\mathbb{Z}_n$ er et endeligt tallegeme, netop når *n* er et usammensat tal, dvs. et primtal.

For at vise at $\mathbb{Z}_n$ er et tallegeme, netop når *n* er et usammensat tal, dvs. et primtal, skal vi undersøge hvornår tal forskellige fra 0 har et reciprokt element.

Vi starter med at bemærke, at $\mathbb{Z}_n$ *er en talring med nuldivisorer* netop *n* er et sammensat tal. Hvis *n* er et sammensat tal, dvs. $n = a \cdot b$, hvor divisorerne *a* og *b* begge er mindre end *n*. Men da gælder jo netop

$$a \cdot b = 0 \bmod n$$

Hvis talringen $\mathbb{Z}_n$ på den anden side indeholder nuldivisorer findes der altså restklasser *a* og *b* forskellige fra 0, så

$$a \cdot b = 0 \bmod n$$

Men det betyder jo netop at $a \cdot b$ er et multiplum af *n*, dvs.

$$a \cdot b = q \cdot n$$

Ved at bortdividere primfaktorerne i *q* på begge sider, ender vi derfor med en relation af formen:

$$a_1 \cdot b_1 = n$$

hvor faktorerne a_1 og b_1 er mindre end a og b og dermed mindre end *n*. Altså er tallet *n* sammensat.

Men en talring med nuldivisorer kan ikke være et tallegeme, fordi nuldivisorer ikke kan have et reciprokt element!

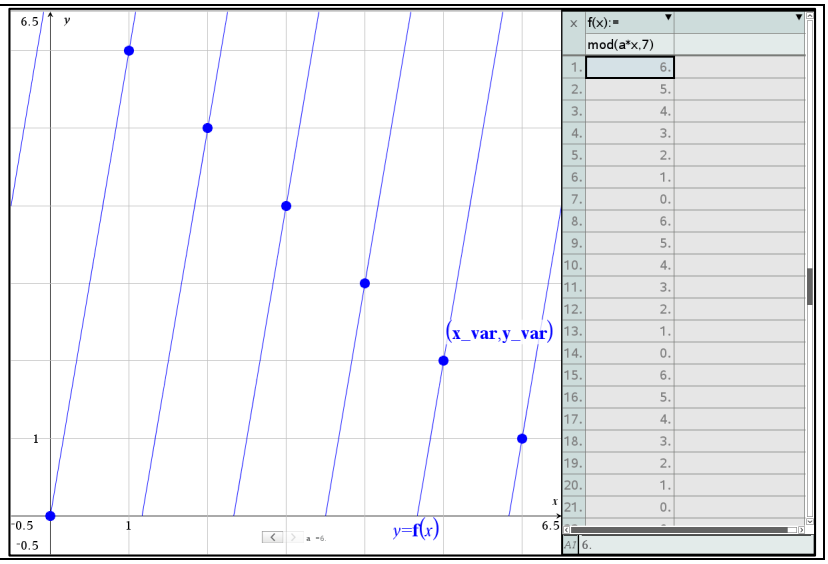
Hvis *n* er sammensat er slaget altså tabt: Restklasseringen $\mathbb{Z}_n$ kan da ikke være et tallegeme.

Vi vender os derfor mod tilfældet hvor *n* er et primtal *p*. Vi skal vise at $\mathbb{Z}_p$ er et tallegeme. Lad nu *a* være en restklasse forskellig fra 0 og se på den lineære funktion $f(x) = a \cdot x \bmod n$.

Øvelse 5: Lineære funktioner modulo 7

a) Tegn graferne for de lineære funktioner modulo 7, dvs. opret en funktionstabel i et regneark og konstruer funktionstabellen for $f(x) = a \cdot x \bmod 7$, hvor *a* er en skydervariabel med værdierne $\{1,2,3,4,5,6\}$.

b) Kommenter graferne! Hvordan kan man fx se af grafen at det reciprokke element til 5 er 6?



Det viser sig nu, at den lineære funktion $f(x) = a \cdot x \bmod n$ er enentydig, dvs. hver *y*-værdi optræder højst én gang! Det skyldes netop at der ikke er nogen nuldivisorer. For hvis $y_1 = y_2$, dvs. $a \cdot x_1 = a \cdot x_2 \bmod n$, slutter vi at der gælder

$$a \cdot x_1 = a \cdot x_2$$
$$a \cdot x_1 - a \cdot x_2 = 0$$
$$a \cdot (x_1 - x_2) = 0$$

Men da *a* ikke er 0, og der ikke findes nuldivisorer kan produktet kun være nul, hvis den anden faktor er 0, dvs. der gælder

$$x_1 - x_2 = 0$$
$$x_1 = x_2$$

Men restklasseringen $\mathbb{Z}_p$ indeholder netop *p* elementer. Og da de højst kan optræde netop én gang som billeder for den lineære funktion $f(x) = a \cdot x \bmod n$, må de alle optræde netop én gang! Der findes altså et *x*, så $a \cdot x = 1$, dvs. $x = a^{-1}$. Hvordan man så finder det reciprokke element i praksis er en helt anden sag. Vi har vist at det findes og det er nok!

Vi har nu fundet et stort reservoir af eksempler på endelige tallegemer, nemlig restklasselegemerne $\mathbb{Z}_p$ modulo et primtal *p*.

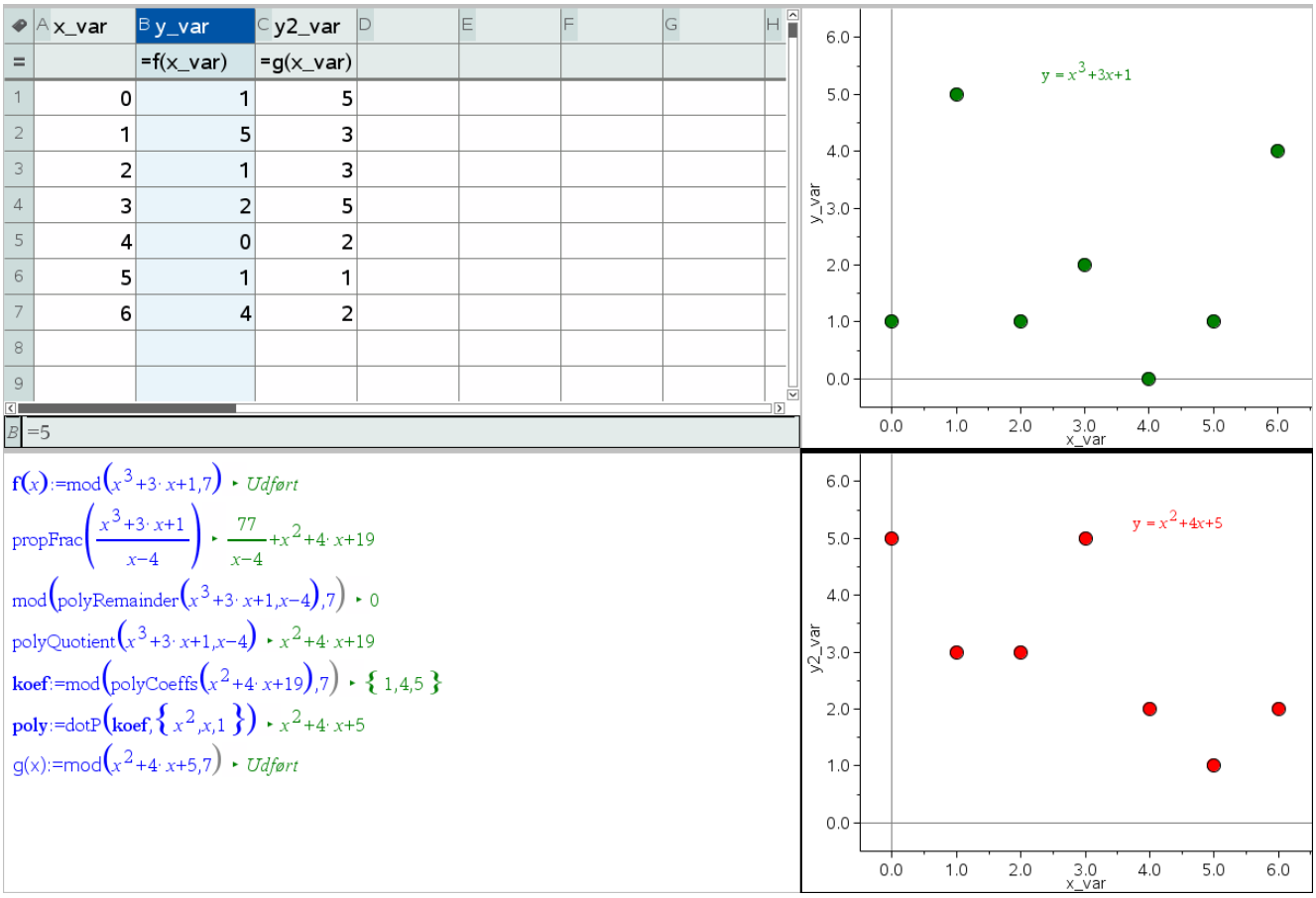
Vi har specielt fundet restklasselegemet $\mathbb{Z}_2 = \{0,1\}$, hvor restklassen 0 repræsenterer alle de lige tal (hvor 2 jo går op) og restklassen 1 repræsenterer alle de ulige tal (hvor 2 netop ikke går op). Regning i $\mathbb{Z}_2$ svarer altså netop til regning med pariteter.

Der findes imidlertid andre endelige tallegemer end restklasselegemerne $\mathbb{Z}_p$. De blev fundet af Galois og kaldes derfor for Galois-legemer. Men før vi kan konstruere Galois-legemerne, skal vi først lære endnu et konstruktionsprincip.

Inden da kigger vi lige kort på polynomier over restklasselegemet $\mathbb{Z}_p$.

Lad os fx se på tredjegradspolynomiet $f(x)=x^3+3x+1 \mod 7$. I et regneark kan vi nemt konstruere en funktionstabel og dermed kigge på grafen som et punktplot. Det giver os fx mulighed for a se efter eventuelle rødder, dvs. skæringer med *x*-aksen. Da grafen er diskret, kan vi derimod *ikke* anvende fx differentialregning til at analysere grafens forløb .

Vi ser da at grafen har netop et nulpunkt, nemlig $x=4$



Men det betyder jo at tredjegradspolynomiet kan faktoriseres i førstegradspolynomiet $x-4\equiv x+3$ og et andengradspolynomium som vi kan finde ved polynomiers division:

Hvis vi udfører divisionen indenfor de rationale tal $\mathbb{Q}$ finder vi:

$$\frac{x^3+3x+1}{x-4}=\frac{77}{x-4}+x^2+4x+19$$

Men vi arbejder jo indenfor tallegemet $\mathbb{Z}_7$ så $77\equiv 0$! Ydermere gælder der $19\equiv 5$, så indenfor tallegemet $\mathbb{Z}_7$ gælder der

$$\frac{x^3+3x+1}{x-4}\equiv x^2+4x+5$$

Vi har også tegnet punktgrafen for andengradspolynomiet og kan netop se at det ikke har nogen rødder, så det kan ikke faktoriseres yderligere.

Øvelse 6:

a) Undersøg nu selv et tilfældigt polynomium af grad højst 5 på samme måde. Vælg først en tilfældig grad mellem 2 og 5 og vælg derefter tilfældige koefficienter fra 0 til 6 til polynomiet.

b) Hvis det kan faktoriseres, så gennemfør faktoriseringen som beskrevet ovenfor.

Restklasselegemerne modulo et primtal udgør et særdeles stærkt redskab indenfor talteori til at udlede simple egenskaber ved primtal. Men det er et helt andet projekt.

3. Fra legeme til ring: Regning med polynomier

3.1 Polynomiumsringen $\mathbb{F}[x]$, hvor $\mathbb{F}$ er et tallegeme

Hvis vi kigger på et tallegeme, fx de rationale tal $\mathbb{Q}$, eller de reelle tal $\mathbb{R}$, så kan vi definere polynomier af grad *n* på den sædvanlige måde som funktioner med forskriften $p(X)=a_0+a_1\cdot x+a_2\cdot x^2+...+a_n\cdot x^n$ med $a_n\neq 0$, hvor koefficienterne $a_0,a_1,a_2,...,a_n$ alle kommer fra tallegemet. Der findes også en rent algebraisk måde at definerer polynomier på som rene algebraiske objekter, men det får vi ikke brug for her.

Mængden af alle polynomier betegnes med $\mathbb{Q}[x]$, hvis der er tale om rationale polynomier, $\mathbb{R}[x]$, hvis der er tale om reelle polynomier osv. Sådanne polynomier kan opfattes som generaliseringer/udvidelser af det underliggende legeme, idet vi kan identificere det underliggende legeme med de *konstante* polynomier, dvs. polynomier med grad 0. Teknisk set har nul-polynomiet $p(x)=0$ dog ingen grad, idet højestegradskoefficienten ikke er forskellig fra 0.

Det er klart at vi kan lægge polynomier sammen, ligesom vi kan gange dem sammen efter de sædvanlige regneregler. Det kan derfor ikke komme som en overraskelse at der gælder følgende sætning:

Sætning 6:
Mængden af polynomier over et tallegeme $\mathbb{F}$ udgør en ring, kaldet polynomiumsringen $\mathbb{F}[x]$.

Bemærkning: På engelsk hedder et legeme a field.

Fx har vi polynomiumsringen over de rationale tal $\mathbb{Q}[x]$, ligesom vi har polynomiumsringen over de reelle tal $\mathbb{R}[x]$. Nulpolynomiet er den konstante funktion 0, et-polynomiet det konstante polynomium 1. Hvis $p(x)=a_0+a_1\cdot x+a_2\cdot x^2+...+a_n\cdot x^n$ så er det modsatte polynomium givet ved polynomiet, hvor vi skifter fortegn på alle koefficienterne, dvs. $-p(x)=-a_0-a_1\cdot x-a_2\cdot x^2-...-a_n\cdot x^n$ osv.

Men polynomiumsringen er *ikke* et tallegeme! Fx er det nemt at indse at identitetspolynomiet *x ikke* har et reciprokt element. For i så fald skulle der gælde

$$x\cdot p(x)=1$$

Men sætter vi $x=0$ fås heraf

$$0=1$$

Hvilket er en modstrid!

Spørgsmålet er så om vi kan omdanne det til et tallegeme på simpel vis? Svaret er bekræftende og vi kan bruge præcis den samme ide, som vi brugte da vi omdannede heltalsringen $\mathbb{Z}$ til et tallegeme, ved at gå over til at regne på restklasser Det er en gammel ide, som bl.a. har været udnyttet af Cauchy til at definere de komplekse tal, så det vil også være et af vores hovedeksempler. Men først ser vi lige kort på division med polynomier. Pointen er nemlig at polynomiumsringen tillader en simpel divisionsalgoritme:

3.1.1 Restklasser med polynomier

Hvis $p(x)$ er et polynomium og $d(x)$ et divisor polynomium kan vi ved hjælp af polynomiers division finde kvotientpolynomiet $q(x)$ og et restpolynomium $r(x)$ med lavere grad end divisorpolynomiet, så der gælder:

$$p(x)=q(x)\cdot d(x)+r(x)$$

Divisionsalgoritmen kan også skrives på formen

$$\frac{p(x)}{d(x)}=q(x)+\frac{r(x)}{d(x)}$$

CAS-værktøjer har normalt både værktøjer til at udføre polynomiers division og finde resten direkte. Det kan fx se således ud:

$$\text{propFrac}\left(\frac{x^3-3\cdot x+1}{x^2-1}\right)\blacktriangleright x-\frac{2\cdot x-1}{x^2-1}$$

$$\text{polyRemainder}\left(x^3-3\cdot x+1,x^2-1\right)\blacktriangleright 1-2\cdot x$$

Af den første divisionsligning fremgår fx at kvotientpolynomiet er $q(x)=x$ og at restpolynomiet er

$$r(x)=-(2\cdot x-1)=1-2\cdot x\,.$$

Af den sidste fremgår kun restpolynomiet. Den sidste kommando svarer til modulus funktionen for de hele tal . Med lidt tålmodighed kan man også finde dem ved håndregning, også selv om man ikke lige har algoritmen for polynomiers division present. Vi udnytter da, at der gælder

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

og finder nu

$$\begin{aligned}\frac{x^3-3\cdot x+1}{x^2-1}&=\frac{\left(x\cdot(x^2-1)+x\right)-3\cdot x+1}{x^2-1}\\&=\frac{x\cdot(x^2-1)}{x^2-1}+\frac{x-3\cdot x+1}{x^2-1}\\&=x+\frac{-2\cdot x+1}{x^2-1}\end{aligned}$$

Men i det følgende bruger vi skamløst CAS-værktøjet til at udføre de mere komplicerede polynomiers divisioner-

Vi kan så arbejde med restklasser præcis lige som vi gjorde det med heltalsringen $\mathbb{Z}$.

Definition 4: Restklasserne modulo d(x)

Lad nu $d(x)$ være et polynomium af grad mindst 1 over et legeme $\mathbb{L}$. Indenfor polynomiumsringen $\mathbb{L}[x]$ kigger vi da på resterne ved division med $d(x)$. De udgør talsystemet $\mathbb{L}[x]/d(x)$ (også kaldet *kvotientringen* modulo $d(x)$).

Læg mærke til at alle resterne $r(x)$ har grad mindre end graden af divisorpolynomiet $d(x)$.

Vi skynder os at se på to konkrete cases, der begge er historisk meget berømte:

CASE 1: $\sqrt{2}$ Irrationale tallegemer

Vi arbejder med polynomier over de rationale tal. Indenfor de rationale tal har andengradspolynomiet $d(x)=x^2-2$ ingen rødder, dvs. vi kan *ikke* løse ligningen $x^2-2=0$ indenfor de rationale tal. Vi kan derfor *ikke* opløse dette andengradspolynomium i faktorer. Vi siger polynomiet er *usammensat* eller *irreducibelt*. Vi danner nu kvotientringen

$$\mathbb{Q}[x]/(x^2-2)$$

Vi ser altså på alle resterne af rationale polynomier ved division med $d(x)=x^2-2$. De består altså af alle polynomier med grad højst 1, dvs. de konstante polynomier og de lineære polynomier. De kan altså skrives på formen

$$r(x)=a+b\cdot x$$

Her er begge koefficienterne rationale tal – og de må gerne være 0.

Vi lægger dem sammen og trækker dem fra hinanden på sædvanlig vis:

$$(a_1+b_1\cdot x)+(a_2+b_2\cdot x)=(a_1+a_2)+(b_1+b_2)\cdot x$$
$$(a_1+b_1\cdot x)-(a_2+b_2\cdot x)=(a_1-a_2)+(b_1-b_2)\cdot x$$

Her er der ingen problemer. Men når vi ganger dem sammen kan vi nemt risikere at graden af produktet bliver 2, og så skal vi reducere graden, dvs. finde resten ved division med $d(x)=x^2-2$. Det er faktisk rimeligt simpelt:

$$\begin{aligned}(a_1+b_1\cdot x)\cdot(a_2+b_2\cdot x)&=(a_1\cdot a_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x+(b_1\cdot b_2)\cdot x^2\\&=(a_1\cdot a_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x+(b_1\cdot b_2)\cdot(x^2-2+2)\\&=(a_1\cdot a_2+2\cdot b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x+(b_1\cdot b_2)\cdot(x^2-2)\\&=(b_1\cdot b_2)\cdot(x^2-2)+((a_1\cdot a_2+2\cdot b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x)\end{aligned}$$

Men her viser divisionsligningen jo at resten er givet ved

$$r(x)=(a_1\cdot a_2+2\cdot b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x$$

Når vi ganger to førstegradspolynomier sammen sker det altså ved hjælp af reglen

$$(a_1+b_1\cdot x)\cdot(a_2+b_2\cdot x)\equiv(a_1\cdot a_2+2\cdot b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_1)\cdot x$$

Her har vi brugt ækvivalenstegnet $\equiv$ for at minde os om at produktet er reduceret med polynomiet $d(x)=x^2-2$!

Der kommer nu to behagelige overraskelser:

Første overraskelse: Når vi bruger det ovenstående produkt på mængden af polynomier af grad højst 1, dvs. konstante polynomier og lineære polynomier, så udgør de et tallegeme!

Vi skal vise at ethvert polynomium $a_0+b_0\cdot x$, bortset fra nul-polynomiet, har et reciprokt element, dvs. vi skal løse ligningssystemet:

$$(a_0+b_0\cdot x)\cdot(a+b\cdot x)\equiv(a_0\cdot a+2\cdot b_0\cdot b)+(a_0\cdot b+a\cdot b_0)\cdot x\equiv 1$$

Det kan omformes til

$$\begin{aligned}a_0\cdot a+2\cdot b_0\cdot b&=1\\b_0\cdot a+\quad a_0\cdot b&=0\end{aligned}$$

Her er determinanten givet ved

$$a_0^{\,2}-2\cdot b_0^{\,2}$$

Men da a_0 og b_0 ikke begge kan være nul ved vi at $a_0^{\,2}-2\cdot b_0^{\,2}$ heller ikke kan være nul – det er oplagt, hvis en af dem er 0, og hvis de begge er forskellige fra 0 ville vi i sidste instans kunne finde et rationalt tal med kvadratet 2! Ligningssystemet har altså netop én løsning og dermed har polynomiet $a_0+b_0\cdot x$ netop et reciprokt element. Det er heller ikke svært at finde løsningen der er givet ved

$$(a_0+b_0\cdot x)^{-1}=\frac{a_0-b_0\cdot x}{a_0^{\,2}-2\cdot b_0^{\,2}}=\left(\frac{a_0}{a_0^{\,2}-2\cdot b_0^{\,2}}\right)+\left(\frac{-b_0}{a_0^{\,2}-2\cdot b_0^{\,2}}\right)\cdot x$$

Anden overraskelse: Indenfor kvotientlegemet kan vi godt løse andengradsligningen $x^2-2=0$!

Der gælder nemlig

$$x^2 - 2 \equiv 0$$

$$x^2 \equiv 2$$

Identitetspolynomiet x har altså netop kvadratet 2, dvs. det spiller rollen som tallet $\sqrt{2}$. Der er derfor *indenfor kvotientlegemet* tradition for simpelthen at kalde identitetspolynomiet *x* for $\sqrt{2}$.

Kvotientlegemet består derfor af alle tal på formen $a + b \cdot \sqrt{2}$, hvor koefficienterne *a* og *b* er rationale tal.

Regnereglerne for disse tal følger da automatisk af de sædvanlige regler når blot vi husker på at der gælder $\left(\sqrt{2}\right)^2 = 2$:

$$\left(a_1 + b_1 \cdot \sqrt{2}\right) + \left(a_2 + b_2 \cdot \sqrt{2}\right) = \left(a_1 + a_2\right) + \left(b_1 + b_2\right) \cdot \sqrt{2}$$

$$\left(a_1 + b_1 \cdot \sqrt{2}\right) - \left(a_2 + b_2 \cdot \sqrt{2}\right) = \left(a_1 - a_2\right) + \left(b_1 - b_2\right) \cdot \sqrt{2}$$

$$\left(a_1 + b_1 \cdot \sqrt{2}\right) \cdot \left(a_2 + b_2 \cdot \sqrt{2}\right) = \left(a_1 \cdot a_2 + 2 \cdot b_1 \cdot b_2\right) + \left(a_1 \cdot b_2 + a_2 \cdot b_2\right) \cdot \sqrt{2}$$

$$\frac{1}{a + b \cdot \sqrt{2}} = \frac{a - b \cdot \sqrt{2}}{a^2 - 2 \cdot b^2}$$

På dette tidspunkt kan man faktisk roligt glemme alt om polynomier og bare regne løs! Man siger at vi har udvidet tallegemet de rational tal $\mathbb{Q}$ med kvadratroden af 2, dvs. $\sqrt{2}$, og det udvidede tallegeme betegnes da blot $\mathbb{Q}[\sqrt{2}]$.

Faktisk kan man indføre det udvidede tallegeme helt simpelt ved i stedet at arbejde med de reelle tal! Indenfor de reelle tal har polynomiet $x^2 - 2$ to rødder $\pm\sqrt{2}$ og kan faktoriseres som

$$x^2 - 2 = \left(x - \sqrt{2}\right) \cdot \left(x + \sqrt{2}\right)$$

Så indenfor de reelle tal er der slet ingen grund til at udvide tallegemet. Her vil man i stedet gå således frem. Det mindste tallegeme indenfor de reelle tal $\mathbb{R}$ er netop de rationale tal $\mathbb{Q}$. Ethvert tallegeme indenfor $\mathbb{R}$ indeholder 0 og 1 og dermed også de hele tal $\mathbb{Z}$, for at være lukket overfor addition og subtraktion og dernæst de rationale tal $\mathbb{Q}$, for også at være lukket over multiplikation og division!

Indenfor de reelle tal udvider vi nu de rationale tal $\mathbb{Q}$ ved at tilføje det reelle tal $\sqrt{2} = 1.414214...$. Det udvidede tallegeme må da i det mindste indeholde alle tallene på formen $a + b \cdot \sqrt{2}$, hvor *a* og *b* er rationale tal. Men da denne talmængde er et tallegeme er udvidelseslegemet, der indeholder $\sqrt{2}$ altså netop givet ved

$$Q[\sqrt{2}] = \left\{a + b \cdot \sqrt{2} \mid a, b \text{ er vilkårlige rationale tal}\right\}$$

Der gælder da oplagt

$$\mathbb{Q} \subset \mathbb{Q}[\sqrt{2}] \subset \mathbb{R}$$

Men selv om vi slet ikke kendte de reelle tal kunne vi altså stadigvæk indføre udvidelseslegemet $\mathbb{Q}[\sqrt{2}]$ ved hjælp af polynomier som beskrevet ovenfor

Øvelse 7: Det gyldne snit

Indenfor de reelle tal er det gyldne snit defineret som tallet $\Phi = \frac{1 + \sqrt{5}}{2} = 1.618...$, der løser andengradsligningen $x^2 - x - 1 = 0$. Men indenfor de rationale tal har denne anden gradsligning ingen rødder. VI kan derfor bruge dette andengradspolynomium til at konstruere en udvidelse af de rationale tals legeme, der også omfatter det gyldne snit Φ. Som ovenfor defineres det som mængden af polynomier af grad højst 1, dvs. konstante funktioner og lineære funktioner med rationale koefficienter:

$$\mathbb{Q}[\Phi] = \left\{a + b \cdot x \mid a, b \text{ er vilkårlige rationale tal}\right\}$$

a) Opstil regnereglerne for summer og produkter af disse polynomier, idet produkterne reduceres ved polynomiers division med $d(x)=x^2-x-1=0$.

b) Gør rede for at identitetspolynomiet *x* har det reciprokke polynomium $x-1$.

c) Gør rede for at ethvert polynomium af grad højst 1 har et reciprokt polynomium indenfor dette talsystem, dvs. der er tale om tallegeme.

d) Gør rede for at hvis vi kalder identitetspolynomiet *x* for Φ kan vi regne på helt normal vis, idet vi blot skal huske på at der må gælde regnereglen $\Phi^2=\Phi+1$.

CASE 2: $\sqrt{-1}$ (De komplekse tal)

De komplekse tal har en lang og fascinerende historie bag sig. Der er mange måder at konstruere de komplekse tal. Oprindeligt blev de konstrueret rent geometrisk ved at forsyne talplanen med to geometriske regneoperationer: addition (i form af parallelforskydninger), og multiplikation (i form af ligedannetheder, dvs. strækninger og rotationer). Den geometriske konstruktion skyldes Wessel, Argand og Gauss. Der findes også en rent algebraisk konstruktion, der stammer fra Hamilton, som også indførte kvaternionerne. Endelig findes der polynomiemetoden, der går tilbage til Cauchy. Det er polynomie-metoden vi her skal se nærmere på, men de tre forskellige metoder fører selvfølgelig frem til præcis de samme komplekse tal

Vi arbejder denne gang med polynomier over de reelle tal. Indenfor de reelle tal har andengradspolynomiet $d(x)=x^2-2$ nu rødder, dvs. vi kan *denne gang* løse ligningen $x^2-2=0$ indenfor de reelle tal. Løsningerne er givet ved $x=\pm\sqrt{2}=\pm 1.414213...$ Vi kan derfor opløse dette andengradspolynomium i to førstegradsfaktorer $x^2-2=\left(x-\sqrt{2}\right)\cdot\left(x+\sqrt{2}\right)$. Vi siger at polynomiet $d(x)=x^2-2$ er *sammensat* eller *reducibelt* over de reelle tal. Vi kan derfor godt nok danne kvotientringen

$$\mathbb{R}[x]/(x^2-2)$$

Men den vil få nuldivisorer, nemlig førstegradspolynomierne $\left(x-\sqrt{2}\right),\left(x+\sqrt{2}\right)$, dvs. der er ingen chance for at det bliver et nyt tallegeme.

Ser vi i stedet på andengradspolynomiet $d(x)=x^2+1$ har det *ingen* reelle rødder. Dette polynomium er altså *usammensat* eller *irreducibelt* over de reelle tal. Det giver derfor god mening at se på alle resterne af rationale polynomier ved division med $d(x)=x^2+1$. De består altså af alle polynomier med grad højst 1, dvs. de konstante polynomier og de lineære polynomier. De kan altså skrives på formen

$$r(x)=a+b\cdot x$$

Her er begge koefficienterne denne gang reelle tal – og de må gerne være 0.

Vi lægger dem sammen og trækker dem fra hinanden på sædvanlig vis:

$$\left(a_1+b_1\cdot x\right)+\left(a_2+b_2\cdot x\right)=\left(a_1+a_2\right)+\left(b_1+b_2\right)\cdot x$$
$$\left(a_1+b_1\cdot x\right)-\left(a_2+b_2\cdot x\right)=\left(a_1-a_2\right)+\left(b_1-b_2\right)\cdot x$$

Her er der ingen problemer. Men når vi ganger dem sammen kan vi nemt risikere at graden af produktet bliver 2, og så skal vi reducere graden, dvs. finde resten ved division med $d(x)=x^2+1$. Det er faktisk rimeligt simpelt:

$$\begin{aligned}(a_1+b_1\cdot x)\cdot(a_2+b_2\cdot x)&=(a_1\cdot a_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x+(b_1\cdot b_2)\cdot x^2\\&=(a_1\cdot a_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x+(b_1\cdot b_2)\cdot(x^2+1-1)\\&=(a_1\cdot a_2-b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x+(b_1\cdot b_2)\cdot(x^2+1)\\&=(b_1\cdot b_2)\cdot(x^2+1)+((a_1\cdot a_2-b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x)\end{aligned}$$

Men her viser divisionsligningen jo at resten er givet ved

$$r(x)=(a_1\cdot a_2-b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_2)\cdot x$$

Når vi ganger to førstegradspolynomier sammen sker det altså ved hjælp af reglen

$$(a_1+b_1\cdot x)\cdot(a_2+b_2\cdot x)\equiv(a_1\cdot a_2-b_1\cdot b_2)+(a_1\cdot b_2+a_2\cdot b_1)\cdot x$$

Her har vi brugt ækvivalenstegnet $\equiv$ for at minde os om at produktet er reduceret med polynomiet $d(x)=x^2+1$!

Der kommer nu to behagelige overraskelser:

Første overraskelse: Når vi bruger det ovenstående produkt på mængden af polynomier af grad højst 1, dvs. konstante polynomier og lineære polynomier, så udgør de et tallegeme!

Vi skal vise at ethvert polynomium $a_0+b_0\cdot x$, bortset fra nul-polynomiet, har et reciprokt element, dvs. vi skal løse ligningssystemet:

$$(a_0+b_0\cdot x)\cdot(a+b\cdot x)\equiv(a_0\cdot a-b_0\cdot b)+(a_0\cdot b+a\cdot b_0)\cdot x\equiv 1$$

Det kan omformes til

$$\begin{aligned}a_0\cdot a-b_0\cdot b&=1\\b_0\cdot a+a_0\cdot b&=0\end{aligned}$$

Her er determinanten givet ved

$$a_0{}^2+b_0{}^2$$

Men da a_0 og b_0 ikke begge kan være nul ved vi at $a_0{}^2+b_0{}^2$ er positiv, dvs. heller ikke kan være nul. Ligningssystemet har altså netop én løsning og dermed har polynomiet $a_0+b_0\cdot x$ netop et reciprokt element. Det er heller ikke svært at finde løsningen der er givet ved

$$(a_0+b_0\cdot x)^{-1}=\frac{a_0-b_0\cdot x}{a_0{}^2+b_0{}^2}=\left(\frac{a_0}{a_0{}^2+b_0{}^2}\right)+\left(\frac{-b_0}{a_0{}^2+b_0{}^2}\right)\cdot x$$

Anden overraskelse: Indenfor kvotientlegemet kan vi godt løse andengradsligningen $x^2+1=0$!

Der gælder nemlig

$$\begin{aligned}x^2+1&\equiv 0\\x^2&\equiv -1\end{aligned}$$

Identitetspolynomiet x har altså netop kvadratet –1, dvs. det spiller rollen som tallet $\sqrt{-1}$. Der er derfor *indenfor kvotientlegemet* tradition for simpelthen at kalde identitetspolynomiet *x* for $\sqrt{-1}$. Der er dog også tradition for at kalde det *i* (for den imaginære enhed).

Kvotientlegemet består derfor af alle tal på formen $a+b\cdot i$, hvor koefficienterne *a* og *b* er reelle tal.

Regnereglerne for disse tal følger da automatisk af de sædvanlige regler når blot vi husker på at der gælder

$$i^2 = \left(\sqrt{-1}\right)^2 = -1:$$

$$\begin{gathered}
(a_1 + b_1 \cdot i) + (a_2 + b_2 \cdot i) = (a_1 + a_2) + (b_1 + b_2) \cdot i \\
(a_1 + b_1 \cdot i) - (a_2 + b_2 \cdot i) = (a_1 - a_2) + (b_1 - b_2) \cdot i \\
(a_1 + b_1 \cdot i) \cdot (a_2 + b_2 \cdot i) = (a_1 \cdot a_2 - b_1 \cdot b_2) + (a_1 \cdot b_2 + a_2 \cdot b_2) \cdot i \\
\frac{1}{a + b \cdot i} = \frac{a - b \cdot i}{a^2 + b^2}
\end{gathered}$$

På dette tidspunkt kan man faktisk roligt glemme alt om polynomier og bare regne løs! Man siger at vi har udvidet tallegemet de reelle tal $\mathbb{R}$ med kvadratroden af –1, dvs. $i = \sqrt{-1}$, og det udvidede tallegeme betegnes da blot de komplekse tals legeme $\mathbb{C} = \mathbb{R}[i]$.

Man kunne forestille sig at man nu kunne gentage spøgen og udvide de komplekse tal ved at finde et tilsvarende simpelt polynomium over de komplekse tal. Men det kan man ikke! Ifølge algebraens fundamentalsætning har ethvert komplekst polynomium mindst én kompleks rod!

I en vis forstand er de komplekse tal derfor det mest omfattende tallegeme der findes (på samme måde som de rationale tal var det mindste tallegeme):

$$\mathbb{Q} \subset \mathbb{R} \subset \mathbb{C}$$

Skal vi udover de komplekse tal skal der derfor ske noget drastisk! Som opdaget af Hamilton kan man opgive kravet om kommutativitet for multiplikationen. Det åbner mulighed for et endnu større tallegeme, kvaternionerne. Hvis man også er villig til at ofte associativiteten for multiplikationen findes der et tallegeme, der rækker ud over kvaternionerne, nemlig oktonionerne. Men så er det også slut! Begge disse tallegeme har vigtige anvendelsesområder:

- Kvaternionerne er uundværlige indenfor moderne avanceret computeranimation (Uden kvaternioner ingen Lara Croft!). Selv om der er tale om firedimensionale objekter er de nemlig fremragende til at styre rotationer i 3-dimensioner (og dermed til at styre flydende bevægelser i computeranuimation)
- Oktonionerne vinder ind indenfor moderne strengteori. Selv om der kun er tale om 8-dimensionale objekter ligger de meget tættere på de 10 rum-dimensioner som man opererer med indenfor strengteorien.

4. Galois-legemerne $\mathbb{GF}\left[p^n\right]$

Vi vender nu tilbage til de endelige tallegemer i form af primtalslegemerne $\mathbb{Z}_p = \{0,1,2,3,\ldots,p-1\}$. Spørgsmålet var nu om der fandtes flere endelige tallegemer end disse? Svaret er bekræftende: De blev fundet af Galois i forbindelse med hans undersøgelser af røddernes opførsel i polynomier med heltalllige koefficienter.

Sætning 7: Galois-legemerne $\mathbb{GF}(p^n)$

For et hvert primtal p og ethvert naturligt tal n findes der et Galois-legeme med netop p^n elementer, der fremkommer som en udvidelse af restklasselegemet $\mathbb{Z}_p$.

Vi vil ikke bevise sætningen men vil konstruere Galois-legemerne i et konkret tilfælde, der efterfølgende vil kunne generaliseres. Vi tager udgangspunkt i paritetetslegemet $\mathbb{Z}_2 = \{0,1\}$, hvor 0 står for de lige tal og 1 står for de ulige

tal. Det er et særligt simpelt tallegeme med regneoperationerne + og ·, der opfylder de følgende yderst simple regnetabeller

+	0	1
0	0	1
1	1	0

·	0	1
0	0	0
1	0	1

Læg mærke til, at der indenfor $\mathbb{Z}_2$ *ikke* er forskel på addition og subtraktion, idet både 0 og 1 er deres eget modsatte tal!

Med udgangspunkt i dette konstruerer vi nu først polynomiumsringen $\mathbb{Z}_2[x]$. Vi skal så have omdannet den til et tallegeme ved at arbejde med restklasser modulo et usammensat, irreducibelt polynomium.

4.1 Irreducible polynomier over $\mathbb{Z}_2$

Vi starter derfor med at kigge efter irreducible polynomier. Vi kan da have glæde af den sædvanlige observation. Udfører vi en polynomiers division med førstegradspolynomiet $x-x_0$ fås

$$p(x)=(x-x_0)\cdot q(x)+r(x)$$

Udregnes værdien for $x=x_0$ fås derfor

$$p(x_0)=0\cdot q(x_0)+r(x_0)=r(x_0)$$

Der gælder altså den sædvanlige sætning:

Sætning 8:
Første gradspolynomiet $x-x_0$ går op i polynomiet $p(x)$ netop når x_0 er en rod, dvs. $p(x_0)=0$.

Det kan vi bruge til at jagte irreducible polynomier over $\mathbb{Z}_2$.
Vi skal da dels tage højde for førstegradspolynomiet *x*, der altså går op i polynomiet $p(x)=a_0+a_1\cdot x+\ldots+a_n\cdot x^n$ netop når $p(0)=a_0=0$. Hvis et polynomium skal have en chance for at være irreducibelt skal konstantleddet altså være 1.

Tilsvarende skal vi tage højde for førstegradspolynomiet $x-1$ (der er det samme som førstegradspolynomiet $x+1$). Det går op i polynomiet $p(x)=a_0+a_1\cdot x+\ldots+a_n\cdot x^n$ netop når $p(1)=a_0+a_1+a_2+\ldots+a_n=0$. Hvis et polynomium skal have en chance for at være irreducibelt, skal summen af koefficienterne altså være 1, dvs. polynomiet skal indeholde et ulige antal led!

Så går jagten ind! Et eventuelt irreducibelt andengradspolynomium skal altså være på formen

$$x^2+x+1$$

Men så er det også irreducibelt, efter som der ikke er andre mulige førstegradsfaktorer.

Sætning 9: Der findes netop ét irreducibelt andengradspolynomium: x^2+x+1

Et eventuelt irreducibelt tredjegradspolynomium skal altså være på formen

$$x^3+x+1 \quad \text{eller} \quad x^3+x^2+1$$

Men hvis det har en andengradsfaktor må der nødvendigvis også være en førstegradsfaktor, og dem har vi udelukket, dvs. der er netop disse to irreducible tredjegradspolynomier.

Sætning 10: Der findes netop to irreducible tredjegradspolynomier: x^3+x+1 og x^3+x^2+1

Et eventuelt irreducibelt fjerdegradspolynomium skal altså være på formen

$$x^4+x+1,\ x^4+x^2+1,\ x^4+x^3+1 \text{ eller } x^4+x^3+x^2+x+1$$

Men denne gang skal vi være mere forsigtige. Vi har udelukket eventuelle førstegradsfaktorer, men vi har ikke udelukket to irreducible andengradsfaktorer! Nu findes der kun et irreducibelt andengradspolynomium. Vi kan altså danne produktet

$$(x^2+x+1)^2 = x^4+x^2+1$$

(idet alle de dobbelte produkter forsvinder automatisk, da 2=0!). Der er altså kun 3 irreducible fjerdegradspolynomier.

Sætning 11: Der findes netop tre irreducible fjerdegradspolynomier: x^4+x+1, x^4+x^3+1 og $x^4+x^3+x^2+x+1$.

Øvelse 8:

a) Prøv nu selv om du kan finde alle de irreducible polynomier af grad 5, 6, 7 og 8!

4.2 Regning med bytes: $\mathbb{GF}\left[2^4\right]$

Når man arbejder med binære koder i computere spiller nu $\mathbb{Z}_2=\{0,1\}$ en særlig rolle, fordi det bygger på de to bits, som al informationen er stykket sammen.

Samler man fire bits fås en *byte*. Vi vil nu se om man tilsvarende kan organisere de 16 bytes som et Galois-legeme, dvs. vi vil konstruere Galois-legement $\mathbb{GF}\left[2^4\right]$! Vi vælger da at opfatte en byte som et polynomium af højst tredje grad:

$$b_3b_2b_1b_0 = b_0+b_1\cdot x+b_2\cdot x^2+b_3\cdot x^3$$

Det er ikke noget problem at lægge dem sammen (hvilket er det samme som at trække dem fra hinanden!). Men når vi ganger sådanne to polynomier sammen bliver graden nemt større end 3 og vi er derfor nødt til at arbejde med restklasser modulo et irreducibelt fjerdegradspolynomium. Vi vælger polynomiet

$$d(x)=x^4+x+1$$

Vi skal så have styr på potenserne af *x*. Vi udarbejder derfor en tabel over alle potenserne idet vi undervejs reducerer potenserne ved at bruge reglen $x^4+x+1\equiv 0$ dvs. $x^4\equiv x+1$.

Øvelse 9:

Starten af tabellen ser således ud:

a) Fuldfør nu selv tabellen. Hvad sker der når vi når frem til x^{15}?

b) Hvorfor viser denne tabel netop at enhver byte forskellige fra 0 har et reciprokt element? Hvad bliver fx det reciprokke element til $x^5=x^2+x$?

1	0
x	1
x^2	2
x^3	3
$x^4=x+1$	4
$x^5=x\cdot x^4=x\cdot(x+1)=x^2+x$	5
…	…
x^{15}	

Vi har nu ikke blot organiseret de 16 bytes som et tallegeme. Vi har også set at ethvert element forskellig fra 0 kan skrives som en potens af identitetspolynomiet *x*. Der er tradition for at give dette navnet α. Enhver byte kan altså skrives på formen

$$b_3b_2b_1b_0 = b_0+b_1\cdot\alpha+b_2\cdot\alpha^2+b_3\cdot\alpha^3, \text{ hvor } \alpha^4\equiv\alpha+1$$

Men vi kan også organisere de 15 bytes forskellig fra 0 efter hvilken eksponent de hører til. Eksponenten kaldes da den diskrete logaritme af byten:

Øvelse 10:

Her ses et udsnit af tabellen

Byte	Repræsentation som potens	Den diskrete logaritme
0000 = 0	–	–
0001 = 1	α^0	0
0010 = α	α^1	1
0011 = α + 1	α^4	4
0100 = α^2	α^2	2
0101 = α^2 + 1		
0110 = α^2 + α	α^5	5
0111 = α^2 + α + 1		
1000 = α^3		
1001 = α^3 + 1		
1010 = α^3 + α		
1011 = α^3 + α + 1		
1100 = α^3 + α^2		
1101 = α^3 + α^2 + 1		
1110 = α^3 + α^2 + α		
1111 = α^3 + α^2 + α + 1		

a) Udfyld nu selv resten af tabellen.

b) Gør rede for hvordan den diskrete logaritme kan bruges til at gange to bytes sammen

Vi har set at polynomiet $d(x)=x^4+x+1$ er irreducibelt over $\mathbb{Z}_2$. Men vi kan også opfatte det som et polynomium over $\mathbb{GF}\left[2^4\right]$! I så fald er α en rod, idet der jo netop gælder $d(\alpha)\equiv\alpha^4+\alpha+1\equiv 0$. Faktisk har fjerdegradspolynomiet nu fire rødder og kan faktoriseres fuldstændigt. De øvrige rødder er α^2, α^4 og α^8. Der gælder nemlig:

$$\begin{aligned} d(\alpha^2) &= \left(\alpha^2\right)^4+\alpha^2+1 \\ &= \left(\alpha^4\right)^2+\alpha^2+1 \\ &= (\alpha+1)^2+\alpha^2+1 \qquad \text{Vi udnytter at } \alpha^4\equiv\alpha+1 \\ &= \alpha^2+1+\alpha^2+1 \qquad \text{Vi udnytter at det dobbelte produkt forsvinder} \\ &= 0 \end{aligned}$$

Øvelse 11:

a) Vis nu at der på samme måde gælder: $d(\alpha^4)=0$ og $d(\alpha^8)=0$.

Vi har altså indenfor $\mathbb{GF}\left[2^4\right]$ fundet en fuldstændig faktorisering af fjerdegradspolynomiet

$$d(x)=x^4+x+1=(x-\alpha)\cdot\left(x-\alpha^2\right)\cdot\left(x-\alpha^4\right)\cdot\left(x-\alpha^8\right)$$

Øvelse 12:

a) Brug gerne støtte fra dit CAS-værktøj til at vise faktoriseringen ved at gange højresiden ud.

4.3 Regning med words: $\mathbb{GF}\left[2^8\right]$

Samler man to *bytes* fås et *word*. Vi vil nu se om man tilsvarende kan organisere de 256 words som et Galois-legeme, dvs. vi vil konstruere Galois-legement $\mathbb{GF}\left[2^8\right]$! Vi vælger da at opfatte et word som et polynomium af højst syvende grad:

$$b_7b_6b_5b_4b_3b_2b_1b_0 = b_0 + b_1 \cdot x + b_2 \cdot x^2 + b_3 \cdot x^3 + b_4 \cdot x^4 + b_5 \cdot x^5 + b_6 \cdot x^6 + b_7 \cdot x^7$$

Det er ikke noget problem at lægge dem sammen (hvilket er det samme som at trække dem fra hinanden!). Men når vi ganger sådanne to polynomier sammen bliver graden nemt større end 7 og vi er derfor nødt til at arbejde med restklasser modulo et irreducibelt ottendegradspolynomium. Vi vælger denne gang polynomiet

$$d(x) = x^8 + x^4 + x^3 + x^2 + 1$$

Vi skal så have styr på potenserne af *x*. Vi udarbejder derfor en tabel over alle potenserne idet vi undervejs reducerer potenserne ved at bruge reglen $x^8 + x^4 + x^3 + x^2 + 1 \equiv 0$ dvs. $x^8 \equiv x^4 + x^3 + x^2 + 1$. Herefter kører alt ligesom ved Galois-legemet $\mathbb{GF}\left[2^4\right]$. Men denne gang er det et meget større regnearbejde, så nu er vi nok nødt til at inddrage computeren aktivt!

Vi viser princippet ved at gennemregne $\mathbb{GF}\left[2^4\right]$ som er nemmere at overskue og som vi allerede har en del erfaring med.

Vi har udvidet $\mathbb{Z}_2$ med roden α i fjerdegradspolynomiet $x^4 + x + 1$. Vi skal have styr på potenserne $1, \alpha, \alpha^2, \alpha^3, \ldots, \alpha^{15}$. Vi opretter derfor et symbolsk regneark og frembringer disse potenser i første søjle. Vi skriver 1 i første celle og ganger derefter med α i de følgende celler, idet vi trækker celleformlen ned gennem regnearket.

I den næste reducerer vi så potenserne ved at udføre en polynomiers division med $\alpha^4 + \alpha + 1$ og beholde resten. Det sker ved hjælp af kommandoen

polyremainder(..., $\alpha^4 + \alpha + 1$)

Derved fås netop en repræsentation af potenserne som polynomier af højst grad 3.

	A	B
=		
1	0	1
2	1	α
3	2	α^2
4	3	α^3
5	4	α^4
6	5	α^5
7	6	α^6
8	7	α^7
9	8	α^8
10	9	α^9
11	10	α^10
12	11	α^11
13	12	α^12
14	13	α^13
15	14	α^14
16	15	α^15
17		
18		
19		
20		
21		

B2 =α· b1

	A	B	C
=			
1	0	1	1
2	1	α	α
3	2	α^2	α^2
4	3	α^3	α^3
5	4	α^4	-α−1
6	5	α^5	-α^2−α
7	6	α^6	-α^3−α^2
8	7	α^7	-α^3+α+1
9	8	α^8	α^2+2*α+1
10	9	α^9	α^3+2*α^2+α
11	10	α^10	2*α^3+α^2−α−1
12	11	α^11	α^3−α^2−3*α−2
13	12	α^12	-α^3−3*α^2−3*α−1
14	13	α^13	-3*α^3−3*α^2+1
15	14	α^14	-3*α^3+4*α+3
16	15	α^15	4*α^2+6*α+3
17			
18			
19			
20			
21			

C1 =polyremainder(b1,α^4+α+1)

Men koefficienterne ligger i heltalsringen $\mathbb{Z}$ og ikke i restklasseringen $\mathbb{Z}_2$. Det er nemt nok at overskue i "hånden": Alle led med lige koefficienter forsvinder, idet et lige tal reducerer til 0. Tilsvarende overlever alle led med ulige koefficienter, idet koefficienten reduceres til 1, idet et ulige tal reducerer til 1.

Men vi skal have gjort det i regnearket, så vi skal have nedbrudt polynomiet til en liste af koefficienter, som vi kan regne på, idet de skal reduceres modulo 2. Det sker ved hjælp af kommandoen

polycoeffs(...,α)

Den frembringer umiddelbart en liste af koefficienter og vi kan *ikke* fremvise en liste i en celle, hvorfor vi er nødt til at konvertere den midlertidigt til en streng .

	A	B	C	D
=				
1	0	1	1	{1}
2	1	α	α	{1,0}
3	2	α^2	α^2	{1,0,0}
4	3	α^3	α^3	{1,0,0,0}
5	4	α^4	-α−1	{1,1}
6	5	α^5	-α^2−α	{1,1,0}
7	6	α^6	-α^3−α^2	{1,1,0,0}
8	7	α^7	-α^3+α+1	{1,0,1,1}
9	8	α^8	α^2+2*α+1	{1,0,1}
10	9	α^9	α^3+2*α^2+α	{1,0,1,0}
11	10	α^10	2*α^3+α^2−α−1	{0,1,1,1}
12	11	α^11	α^3−α^2−3*α−2	{1,1,1,0}
13	12	α^12	-α^3−3*α^2−3*α−1	{1,1,1,1}
14	13	α^13	-3*α^3−3*α^2+1	{1,1,0,1}
15	14	α^14	-3*α^3+4*α+3	{1,0,0,1}
16	15	α^15	4*α^2+6*α+3	{0,0,1}
17				
18				
19				
20				
21				

D1 =string(mod(polycoeffs(c1,α),2))

Derefter skal vi have bygget polynomiet op igen. Det kræver et lille trick undervejs, fordi koefficientlisterne *ikke* har samme længde. Vi skal derfor have tilpasset potenslisten

$\{\alpha^3,\alpha^2,\alpha,1\}$

	A	B	C	D	E
=					
1	0	1	1	{1}	{1}
2	1	α	α	{1,0}	{α,1}
3	2	α^2	α^2	{1,0,0}	{α^2,α,1}
4	3	α^3	α^3	{1,0,0,0}	{α^3,α^2,α,1}
5	4	α^4	-α−1	{1,1}	{α,1}
6	5	α^5	-α^2−α	{1,1,0}	{α^2,α,1}
7	6	α^6	-α^3−α^2	{1,1,0,0}	{α^3,α^2,α,1}
8	7	α^7	-α^3+α+1	{1,0,1,1}	{α^3,α^2,α,1}
9	8	α^8	α^2+2*α+1	{1,0,1}	{α^2,α,1}
10	9	α^9	α^3+2*α^2+α	{1,0,1,0}	{α^3,α^2,α,1}
11	10	α^10	2*α^3+α^2−α−1	{0,1,1,1}	{α^3,α^2,α,1}
12	11	α^11	α^3−α^2−3*α−2	{1,1,1,0}	{α^3,α^2,α,1}
13	12	α^12	-α^3−3*α^2−3*α−1	{1,1,1,1}	{α^3,α^2,α,1}
14	13	α^13	-3*α^3−3*α^2+1	{1,1,0,1}	{α^3,α^2,α,1}
15	14	α^14	-3*α^3+4*α+3	{1,0,0,1}	{α^3,α^2,α,1}
16	15	α^15	4*α^2+6*α+3	{0,0,1}	{α^2,α,1}
17					
18					
19					
20					
21					

E1 =string(right({α^3,α^2,α,1},dim(expr(d1))))

Men så er vi faktisk også igennem. Vi skal nu bare have samlet koefficientlisten (mod 2) med potenslisten og det kan vi fx gøre med et skalarprodukt :

Dermed har vi fået genskabt strukturtabellen for Galoislegemet $\mathbb{GF}\left[2^4\right]$.

Læg mærke til at den første søjle netop angiver den diskrete logaritme til tallene i Galoislegemet, dvs. den sidste søjle.
Læg også mærke til at den sidste række faktisk er overflødig. Den tjener udelukkende til at tjekke at potenserne er cykliske, dvs. det hele gentager sig nu igen. Til gengæld har vi ikke taget 0 med i strukturtabellen. Men 0 er jo heller ikke en potens. Og så er det jo nemt at gange med 0 – dertil behøver man ikke en diskret logaritmetabel

	A	B	C	D	E	F
=						
1	0	1	1	{1}	{1}	1
2	1	α	α	{1,0}	{α,1}	α
3	2	α^2	α^2	{1,0,0}	{α^2,α,1}	α^2
4	3	α^3	α^3	{1,0,0,0}	{α^3,α^2,α,1}	α^3
5	4	α^4	-α−1	{1,1}	{α,1}	α+1
6	5	α^5	-α^2−α	{1,1,0}	{α^2,α,1}	α^2+α
7	6	α^6	-α^3−α^2	{1,1,0,0}	{α^3,α^2,α,1}	α^3+α^2
8	7	α^7	-α^3+α+1	{1,0,1,1}	{α^3,α^2,α,1}	α^3+α+1
9	8	α^8	α^2+2*α+1	{1,0,1}	{α^2,α,1}	α^2+1
10	9	α^9	α^3+2*α^2+α	{1,0,1,0}	{α^3,α^2,α,1}	α^3+α
11	10	α^10	2*α^3+α^2−α−1	{0,1,1,1}	{α^3,α^2,α,1}	α^2+α+1
12	11	α^11	α^3−α^2−3*α−2	{1,1,1,0}	{α^3,α^2,α,1}	α^3+α^2+α
13	12	α^12	-α^3−3*α^2−3*α−1	{1,1,1,1}	{α^3,α^2,α,1}	α^3+α^2+α+1
14	13	α^13	-3*α^3−3*α^2+1	{1,1,0,1}	{α^3,α^2,α,1}	α^3+α^2+1
15	14	α^14	-3*α^3+4*α+3	{1,0,0,1}	{α^3,α^2,α,1}	α^3+1
16	15	α^15	4*α^2+6*α+3	{0,0,1}	{α^2,α,1}	1

Skal vi gange to galoistal sammen, kan vi derfor nøjes med at slå deres diskrete logaritmer op, lægge dem sammen (modulo 15), og så igen slå op hvilket galoistal, der svarer til summen

Øvelse 13

a) Frembring nu selv en strukturtabel for Galoislegemet $\mathbb{GF}\left[2^8\right]$

b) Eftervis at rødderne i generatorpolynomiet $x^8+x^4+x^3+x^2+1$ netop er givet ved $\alpha,\alpha^2,\alpha^4,\alpha^8,\alpha^{16},\alpha^{32},\alpha^{64},\alpha^{128}$.

4.4 Regning i Galoislegemerne $\mathbb{GF}\left[2^n\right]$

Når man skal arbejde med fejlrettende koder skal man arbejde i diverse Galoislegemer. Vi har set på de to hovedtilfælde 4-bit koder og 8-bit koder, men man kan sagtens møde andre tilfælde. Vi starter derfor med en oversigt over passende irreducible generatorpolynomier, som man kan bruge for andre forekommende Galoislegemer:

Grad	*Generator*	*Grad*	*Generator*
3	$1+x+x^3$	14	$1+x+x^6+x^{10}+x^{14}$
4	$1+x+x^4$	15	$1+x+x^{15}$
5	$1+x^2+x^5$	16	$1+x+x^3+x^{12}+x^{16}$
6	$1+x+x^6$	17	$1+x^3+x^{17}$
7	$1+x^3+x^7$	18	$1+x^7+x^{18}$
8	$1+x^2+x^3+x^4+x^8$	19	$1+x+x^2+x^5+x^{19}$
9	$1+x^4+x^9$	20	$1+x^3+x^{20}$
10	$1+x^3+x^{10}$	21	$1+x^2+x^{21}$
11	$1+x^2+x^{11}$	22	$1+x+x^{22}$
12	$1+x+x^4+x^6+x^{12}$	23	$1+x^5+x^{23}$
13	$1+x+x^3+x^4+x^{13}$	24	$1+x+x^2+x^7+x^{24}$

De firebits- og ottebits-koder vi hidtil har arbejdet med er fremhævet med gult i skemaet.

4.4.1 Standardrepræsentationerne af Galoislegemer over $\mathbb{Z}_2$.

I det følgende vil vi nu koncentrere os om firebits- og ottebits-koderne, dvs. om Galoislegemerne $\mathbb{GF}\left[2^4\right]$ og $\mathbb{GF}\left[2^8\right]$. Der findes flere forskellige repræsentationer af elementerne i sådanne Galoislegemer. Der er 16 elementer i $\mathbb{GF}\left[2^4\right]$ så det er nærliggende at repræsentere dem med tallene $\{0,1,2,3,...,15\}$ men der er *ikke* tale om restklasseringen $\mathbb{Z}_{16}$. Så vi kan ikke lægge dem sammen eller gange dem med hinanden sådan som vi ville gøre med restklasser. Når vi regner med dem skal det ske efter reglerne for regning med polynomier!

Vi kan omsætte dem til binære tal, hvilket er tættere på polynomierne, fx

$$11_{dec} = 8+2+1 = 1011_{bin}$$

I så fald er de binære cifre netop koefficienterne i polynomiet, dvs. polynomiumsrepræsentationen er tredjegradspolynomiet

$$1\cdot x^3+0\cdot x^2+1\cdot x^1+1\cdot x^0 = x^3+x+1$$

Da vi typisk kalder identitetspolynomiet *x* for α kan dette også skrives

$$\alpha^3+\alpha+1$$

Endelig har vi set at alle elementerne bortset fra 0 kan skrives som potenser af α, i dette tilfælde altså α^7, jfr. strukturtabellen side 27.

Men addition indenfor $\mathbb{Z}_2$ er formelt det samme som xor-operationen (exclusive or, dvs. den ene er sand, den anden er falsk):

+	0	1
0	0	1
1	1	0

xor	False	True
False	False	True
True	True	False

Xor	0	1
0	0	1
1	1	0

Den logiske operation *xor* er så fundamental at den også virker på heltal i CAS-programmer. Vi kan altså uden videre lægge tal sammen i Galois-legemerne idet vi blot skal huske på at bruge xor-operationen til at udføre additionen:

```
11 xor 5 ▸ 14
112 xor 57 ▸ 73
```

Her kan man forestille sig den øverste addition udført i $\mathbb{GF}\left[2^4\right]$ og den nederste i $\mathbb{GF}\left[2^8\right]$.

Det er lidt mere kompliceret at gange elementer fra Galois-legemer sammen, idet vi her skal bruge polynomie-multiplikation efterfulgt af en polynomiers division med generator-polynomiet for Galois-legemet. Det gør det sværere fx at gange to elementer sammen, hvis de er repræsenteret som heltal. Det er her man i praksis bruger den diskrete logaritme-tabel, der lægges ind i computeren. Udgangspunktet er strukturtabellen, hvor vi tilføjer heltalsrepræsentationen, der fås fra polynomie-repræsentationen ved midlertidigt at sætte $\alpha = 2$:

	A	B	C	D	E	F	G
=							
1	0	1	1	{1}	{1}	1	1
2	1	α	α	{1,0}	{α,1}	α	2
3	2	α^2	α^2	{1,0,0}	{α^2,α,1}	α^2	4
4	3	α^3	α^3	{1,0,0,0}	{α^3,α^2,α,1}	α^3	8
5	4	α^4	-α−1	{1,1}	{α,1}	α+1	3
6	5	α^5	-α^2−α	{1,1,0}	{α^2,α,1}	α^2+α	6
7	6	α^6	-α^3−α^2	{1,1,0,0}	{α^3,α^2,α,1}	α^3+α^2	12
8	7	α^7	-α^3+α+1	{1,0,1,1}	{α^3,α^2,α,1}	α^3+α+1	11
9	8	α^8	α^2+2*α+1	{1,0,1}	{α^2,α,1}	α^2+1	5
10	9	α^9	α^3+2*α^2+α	{1,0,1,0}	{α^3,α^2,α,1}	α^3+α	10
11	10	α^10	2*α^3+α^2−α−1	{0,1,1,1}	{α^3,α^2,α,1}	α^2+α+1	7
12	11	α^11	α^3−α^2−3*α−2	{1,1,1,0}	{α^3,α^2,α,1}	α^3+α^2+α	14
13	12	α^12	-α^3−3*α^2−3*α−1	{1,1,1,1}	{α^3,α^2,α,1}	α^3+α^2+α+1	15
14	13	α^13	-3*α^3−3*α^2+1	{1,1,0,1}	{α^3,α^2,α,1}	α^3+α^2+1	13
15	14	α^14	-3*α^3+4*α+3	{1,0,0,1}	{α^3,α^2,α,1}	α^3+1	9
16	15	α^15	4*α^2+6*α+3	{0,0,1}	{α^2,α,1}	1	1
17							
18							
19							
20							
21							

G1 =f1|α=2

Vi giver nu de forskellige repræsentationer navne (diskret logaritme, diskret eksponential, binær, polynomial og decimal (dvs. heltal). Vi samler dem i et særskilt regneark og fjerne den sidste overflødige række:

	A d_log	B d_eks	C bin	D poly	E dec
=					
1	0	1	{1}	1	1
2	1	α	{1,0}	α	2
3	2	α^2	{1,0,0}	α^2	4
4	3	α^3	{1,0,0,0}	α^3	8
5	4	α^4	{1,1}	α+1	3
6	5	α^5	{1,1,0}	α^2+α	6
7	6	α^6	{1,1,0,0}	α^3+α^2	12
8	7	α^7	{1,0,1,1}	α^3+α+1	11
9	8	α^8	{1,0,1}	α^2+1	5
10	9	α^9	{1,0,1,0}	α^3+α	10
11	10	α^10	{0,1,1,1}	α^2+α+1	7
12	11	α^11	{1,1,1,0}	α^3+α^2+α	14
13	12	α^12	{1,1,1,1}	α^3+α^2+α+1	15
14	13	α^13	{1,1,0,1}	α^3+α^2+1	13
15	14	α^14	{1,0,0,1}	α^3+1	9

I denne tabel kan vi nu aflæse de andre repræsentationer, hvis vi kender logaritmen, idet det er logaritmesøjlen, der er ordnet i naturlig rækkefølge. Det er altså en eksponentialtabel.

Sorterer vi i stedet efter heltalsrepræsentationen fås den viste logaritmetabel. Grundtallet er 2 eller α.

	A dec	B d_log	C d_eks	D bin	E poly
=					
1	1	0	1	{1}	1
2	2	1	α	{1,0}	α
3	3	4	α^4	{1,1}	α+1
4	4	2	α^2	{1,0,0}	α^2
5	5	8	α^8	{1,0,1}	α^2+1
6	6	5	α^5	{1,1,0}	α^2+α
7	7	10	α^10	{0,1,1,1}	α^2+α+1
8	8	3	α^3	{1,0,0,0}	α^3
9	9	14	α^14	{1,0,0,1}	α^3+1
10	10	9	α^9	{1,0,1,0}	α^3+α
11	11	7	α^7	{1,0,1,1}	α^3+α+1
12	12	6	α^6	{1,1,0,0}	α^3+α^2
13	13	13	α^13	{1,1,0,1}	α^3+α^2+1
14	14	11	α^11	{1,1,1,0}	α^3+α^2+α
15	15	12	α^12	{1,1,1,1}	α^3+α^2+α+1

Øvelse 14

a) Benyt nu de to ovenstående tabeller til at forklare hvordan man ganger to elementer fra Galois-legemet med hinanden. Begrund fx at der må gælde 5*8 = 14.