RSA-kryptografi för gymnasiet

Jonas Gustafsson & Isac Olofsson

HT 2010

Innehåll

Förord

Denna bok riktar sig till gymnasieelever som vill fördjupa sig i ämnet RSA-kryptografi. RSA-kryptografi är en avancerad metod för att kommunicera med hemliga meddelanden och används flitigt inom t.ex. bankvärlden. När du handlar med ditt kort eller använder din e-legitimation används RSA-kryptografi för att allt du gör ska vara skyddat och säkert. Vid stora transaktioner mellan olika banker används också RSA-kryptografin för att både den som betalar och den som får betalt ska vara säkra att allt går rätt till.

Boken är uppdelad i fyra kapitel. Kapitel 3 och 4 är betydligt mer avancerade än kapitel 1 och 2. Kapitel 1 består mestadels av exempel och övningar som behandlar matematiken som krävs för att kunna utföra RSA-kryptografi med små tal. Kapitel 2 använder matematiken i kapitel 1 för att genom exempel och övingar metodiskt lära ut hur RSA-kryptografi med små tal går till. Kapitel 3 visar matematiken som ligger till grund för att RSA-kryptografi fungerar. Detta visas med hjälp av exempel, satser, förtydligade bevis samt några enstaka övningar. Kapitel 4 förklarar varför RSA-kryptografi är säkert och enkelt att använda. Primtalstester utgör det viktigaste ämnet i detta sista kapitel.

När det gäller hur denna bok ska användas rekommenderas att inte hänga upp sig på smådetaljer. Om du fastnar ska du gå vidare och kolla upp det om det ställer till problem senare. Vi föreslår att kapitel 1 behandlas först och att detta kapitels alla övningar behärskas innan övergång till kapitel 2 sker. Då kapitel 2 är genomfört finns två alternativ. Det första är till för dig, som bara vill lära dig lite mer om RSA-kryptografi och innebär att du hoppar till kapitel 4 och läser det du förstår. Det andra alternativet är för dig som verkligen vill förstå varför RSA-kryptografi fungerar och är säkert. Detta alternativ innebär att du noggrant läser kapitel 3 och kapitel 4.

Kompletterande litteratur

- Agrawal Manindra, Kayal Neeraj, Saxena Nitin (2002), *PRIMES is in P.* Indian Institute of Technology Kanpur
- Axelsson Rolf, Jakobsson Gunilla, Jakobsson Lars, Nilson Klas, Wallin Hans (2002), *Diskret Matematik för gymnasiet.* Liber
- Björk Lars-Eric, Björk Jonas, Brolin Hans (2005), *Matematik 3000: matematik tretusen. Diskret Matematik.* Natur och Kultur

- Björner Anders, *Kryptografi och primalitet.* `http://www.math.kth.se/~bjorner/F1kurs/KryptoPrim.pdf`
- Hellström Lennart, Johansson Per-Gunnar, Morander Staffan, Tengstrand Anders (2001), *Elementär algebra.* Studentlitteratur AB
- Schellwat Holger, Sundhäll Marcus (2009), *RSA-kryptering i enkla steg.* `http://www.oru.se/PageFiles/5818/rsa-talk-09.pdf`
- Singh Simon (1999), *Kodboken.* Norsteds förlag
- Sjögren Jörgen (2000), *Talteori och kryptografi.* `http://www.his.se/PageFiles/17737/Talteori%20och%20krypto.pdf`

I kapitel 1 där vi går igenom olika grundläggande beräkningsmetoder och begrepp har vi använt oss av böckerna *Diskret Matematik för gymnasiet, Matematik 3000: matematik 3000. Diskret Matematik* och *Elementär algebra*. Vi har utifrån dessa böcker skapat oss en uppfattning om vilka grundläggande kunskaper eleverna bör förvärva för att kunna RSA-kryptografera. Med hjälp av dessa böcker och egna erfarenheter har vi konstruerat lämpliga exempel och övningar.

I kapitel 2 har vi arbetat mycket med *Kodboken* för att få en helhetsbild av kryptologi och framförallt RSA-kryptografi. Bland annat redogör *Kodboken* för alla begrepp som förekommer inom RSA-kryptografin. Vi har också hämtat inspiration från *Kryptografi och primalitet* som ger en inledande inblick i RSA-kryptografins olika beräkningsmetoder.

I kapitel 3 som förklarar matematiken bakom RSA-kryptografin, har vi arbetat mycket med *Talteori och kryptografi* för att få en fördjupad kunskap om varför RSA-kryptografin fungerar rent matematiskt.

Slutligen använder vi oss av *Kryptografi och primalitet* och *PRIMES is in P* för att redogöra kring primtalstester i kapitel 4.

Vi kommer i vissa avsnitt att hänvisa till viss litteratur för vidare läsning. All den litteraturen återfinns under kompletterande litteratur.

Kapitel 1

Grundläggande beräkningsmetoder och begrepp

1.1 Mängder

En *mängd* är en samling av föremål eller objekt. De föremål som ingår i mängden kallas *element*. Det finns många exempel på olika typer av mängder, men vi kommer i denna bok bara behandla mängder som har tal som element, dvs *talmängder*. Hur man betecknar en mängd matematisk kan variera, vi kommer dock enbart använda oss av en variant. Låt mängden A bestå av elementen $1, 2, 4, 6$. Vi skriver att

$$A = \{1, 2, 4, 6\}$$

och att exempelvis $2 \in A$, $3 \notin A$, dvs 2 tillhör mängden A men det gör inte 3.

1.2 Kvot och rest

Vi börjar med $29/12 \approx 2{,}41667$. Detta kan också skrivas som *kvot och rest*: $29 = 2 \cdot 12 + 5$. Vi ser att 29 går att dela med 12 två hela gånger, då får vi $2 \cdot 12 = 24$ och resten blir $29 - 24 = 5$. Vi har alltså att kvoten är 2 och resten är 5.

Exempel 1.2.1. Skriv 33 som kvot och rest då 33 divideras med 6.

Lösning: Vi har

$$33/6 = 5{,}5$$
$$5 \cdot 6 = 30$$

och resten blir

$$33 - 30 = 3.$$

Alltså kan vi skriva

$$33 = 5 \cdot 6 + 3.$$

Övning 1. Skriv som kvot och rest då

a) 26 divideras med 11

b) 9 divideras med 3

c) 16 divideras med 3.

Om olika heltal lämnar samma rest vid division med ett visst heltal sägs dessa olika heltal tillhöra samma *restklass*.

Exempel 1.2.2. Vi har att heltalen -5, -2, 1, 4, 7 tillhör samma restklass om de divideras med 3. Detta eftersom

$$7 = 2 \cdot 3 + 1$$
$$4 = 1 \cdot 3 + 1$$
$$1 = 0 \cdot 3 + 1$$
$$-2 = -1 \cdot 3 + 1$$
$$-5 = -2 \cdot 3 + 1.$$

Samtliga dessa lämnar resten 1 vid division med 3. Det finns oändligt många heltal som lämnar rest 1 vid division med 3 eftersom vi kan sätta vilket heltal som helst som kvot (talet efter "är lika med tecknet"). Vi skriver denna restklass som

$$[1]_3 = \{\ldots, -5, -2, 1, 4, 7, \ldots\}.$$

Punkterna i början och slutet av mängden visar att det finns oändligt många tal i restklassen som är mindre än -5 och oändligt många tal i restklassen som är större än 7.

Exempel 1.2.3. Vilka tal finns i restklassen $[3]_6$, det vill säga vilka heltal ger resten 3 då de divideras med 6?

Lösning: Vi har att

$$\begin{aligned} 9 &= 1 \cdot 6 + 3 \\ 3 &= 0 \cdot 6 + 3 \\ -3 &= -1 \cdot 6 + 3. \end{aligned}$$

Vi får

$$[3]_6 = \{\ldots, -3, 3, 9, \ldots\}.$$

Exempel 1.2.4. Vilket tal d i restklassen $[-9]_{12}$ uppfyller villkoret $0 \leq d < 12$?

Lösning: Vi har att

$$\begin{aligned} 15 &= 2 \cdot 12 - 9 \\ 3 &= 1 \cdot 12 - 9 \\ -9 &= 0 \cdot 12 - 9 \end{aligned}$$

$$[-9]_{12} = \{\ldots, -9, 3, 15, \ldots\}.$$

Alltså är $d = 3$.

Observera att det alltid finns endast ett tal d i restklassen $[m]_n$ som uppfyller villkoret $0 \leq d < n$, eftersom avståndet mellan talen i restklassen måste vara n. Skulle det finnas exempelvis två stycken tal c och d som uppfyller villkoret $0 \leq c, d < n$ så skulle avståndet mellan de talen vara mindre än n, vilket ger motsägelse.

Övning 2. Vi har

a) Vilket tal d i restklassen $[-5]_7$ uppfyller villkoret $0 \leq d < 7$?

b) Vilket tal d i restklassen $[-22]_{37}$ uppfyller villkoret $0 \leq d < 37$?

1.3 Delbarhet

De flesta tal kan skrivas som produkter av mindre tal. Till exempel är $14 = 2 \cdot 7$. Om ett heltal a kan skrivas som $a = b \cdot c$, där b och c är heltal så är a *delbart* med b, och a är delbart med c. Talen b och c kallas *delare* till a. Vi skriver att $b|a$ och $c|a$. Att a är delbart med b och c är samma sak som att a ligger i restklasserna $[0]_b$ respektive $[0]_c$, eftersom resten är 0 när a divideras med b eller c. Vi skriver att $a \in [0]_b$ och att $a \in [0]_c$.

Exempel 1.3.1. Bestäm samtliga delare till 15.

Lösning: Vi har

$$a = b \cdot c$$
$$15 = 5 \cdot 3.$$

Alltså är 15 delbart med 5 och 3. Dessutom är 15, liksom alla nollskilda heltal, delbart med sig själv och 1. Vi har alltså att 15, 5, 3 och 1 är samtliga delare till 15.

Övning 3. Bestäm samtliga delare till

a) 24

b) 29.

1.4 Primtal

Ett *primtal* är ett positivt heltal, större än 1, som endast är delbart med 1 och sig självt.

Exempel 1.4.1. Vilka av heltalen $1 - 10$ är primtal?

Lösning: Vi undersöker varje tal för sig:

1 är inget primtal eftersom det inte är större än 1.
2 är ett primtal eftersom det bara är delbart med 1 och sig självt.
3 är ett primtal eftersom det bara är delbart med 1 och sig självt.
4 är inget primtal eftersom det är delbart med 2. Alla jämna tal är delbara med 2 och därmed inga primtal.
5 är ett primtal eftersom det bara är delbart med 1 och sig självt.
6 är inget primtal eftersom det är ett jämnt tal.
7 är ett primtal eftersom det bara är delbart med 1 och sig självt.
8 är inget primtal eftersom det är ett jämnt tal.
9 är inget primtal eftersom det är delbart med 3.
10 är inget primtal eftersom det är ett jämnt tal.

Övning 4. Vilka av heltalen $11 - 15$ är primtal?

1.5 Faktorisering

Vissa tal kan skrivas som produkter av mindre tal på flera sätt. Till exempel kan 24 skrivas både som $6 \cdot 4$ och $8 \cdot 3$. Om vi däremot kräver att delarna är primtal kan varje positivt heltal enbart skrivas som en produkt av primtal på precis ett sätt.

Exempel 1.5.1. Vi har nedan skrivit talen som produkter av primtal.

$$\begin{aligned} 8 &= 2 \cdot 4 = 2 \cdot 2 \cdot 2 = 2^3 \\ 12 &= 2 \cdot 6 = 2 \cdot 2 \cdot 3 = 2^2 \cdot 3 \\ 77 &= 7 \cdot 11 \\ 120 &= 2 \cdot 60 = 2 \cdot 2 \cdot 30 = 2 \cdot 2 \cdot 2 \cdot 15 = 2 \cdot 2 \cdot 2 \cdot 3 \cdot 5 = 2^3 \cdot 3 \cdot 5 \\ 1961 &= 37 \cdot 53. \end{aligned}$$

Ju större det sammansatta talet är desto svårare blir det att dela upp talet i primtalsfaktorer.

1.6 Största gemensamma delare

Det största talet som kan dela två tal kallas den *största gemensamma delaren*.

Exempel 1.6.1. Beräkna största gemensamma delaren för 45 och 63.

Lösning: Vi har

$$\begin{aligned} 45 &= 9 \cdot 5 = 3 \cdot 3 \cdot 5 \\ 63 &= 9 \cdot 7 = 3 \cdot 3 \cdot 7. \end{aligned}$$

Vi ser att den största gemensamma delaren för 45 och 63 blir $3 \cdot 3 = 9$. Detta skriver vi kortare som $\text{sgd}(45,63) = 9$.

Om två tal har största gemensamma delaren 1 är dessa två tal *relativt prima*. Vi skriver att a och b är relativt prima om $\text{sgd}(a,b) = 1$.

Exempel 1.6.2. Bestäm om 15 och 26 är relativt prima.

Lösning: Vi har

$$\begin{aligned} 15 &= 3 \cdot 5 \\ 26 &= 2 \cdot 13. \end{aligned}$$

De har ingen gemensam faktor förutom 1 (som vi inte skriver ut). Alltså är $\text{sgd}(15,26) = 1$ och de är därför relativt prima.

Övning 5. Bestäm största gemensamma delaren för talen och avgör om de är relativt prima.

a) 10 och 21

b) 93 och 62

c) 144 och 90.

1.7 Euklides algoritm

För att bestämma den största gemensamma delaren för två tal kan man använda sig av en metod som heter *Euklides algoritm*.

Exempel 1.7.1. Bestäm sgd(572,196) med hjälp av Euklides algoritm.

Lösning: Vi börjar med att skriva 572 som kvot och rest då 572 divideras med 196.

$$572 = 2 \cdot 196 + 180.$$

Sedan skriver vi 196 som en ny kvot och rest då 196 divideras med ovanstående resten 180.

$$196 = 1 \cdot 180 + 16.$$

Sedan skriver vi 180 som kvot och rest då 180 divideras med ovanstående resten 16. Vi fortsätter på samma sätt tills vi får resten 0.

$$180 = 11 \cdot 16 + 4$$
$$16 = 4 \cdot 4 + 0.$$

Den näst sista resten, i det här fallet 4, är den största gemensamma delaren. Vi har alltså att sgd(572,196) = 4.

Exempel 1.7.2. Avgör med hjälp av Euklides algoritm om 679 och 167 är relativt prima.

Lösning: Vi har

$$679 = 4 \cdot 167 + 11$$
$$167 = 15 \cdot 11 + 2$$
$$11 = 5 \cdot 2 + 1$$
$$2 = 2 \cdot 1 + 0.$$

Den näst sista resten är 1. Alltså är sgd(697,167) = 1, och 679 och 167 är relativt prima.

Övning 6. Avgör med hjälp av Euklides algoritm om talen är relativt prima.

a) 40 och 7

b) 128 och 198

c) 144 och 439.

1.8 Diofantiska ekvationer

Om a, b och c är kända heltal och x och y är okända heltal kallas

$$ax + by = c$$

en *diofantisk ekvation*. Denna kan lösas genom att utföra Euklides algoritm och sedan utföra Euklides algoritm baklänges. I denna bok kommer vi alltid räkna med $c = 1$.

Exempel 1.8.1. Bestäm en lösning för diofantiska ekvationen $5x + 8y = 1$.

Lösning: Vi börjar med att utföra Euklides algoritm på 5 och 8.

$$\begin{aligned} 8 &= 1 \cdot 5 + 3 \\ 5 &= 1 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1 \\ 2 &= 2 \cdot 1 + 0. \end{aligned}$$

Vi ser att sgd(8,5) = 1. (Framöver kommer vi att stanna algoritmen då vi når resten 1). Nu ska vi utföra algoritmen baklänges. Eftersom vi i det tredje ledet ovan har att

$$3 = 1 \cdot 2 + 1$$

kan vi skriva att

$$1 = 1 \cdot 3 - 1 \cdot 2.$$

Med hjälp av andra ledet i Euklides algoritmen ovan skriver vi om 2:an ovan som $2 = 1 \cdot 5 - 1 \cdot 3$. Detta ger att

$$\begin{aligned} 1 &= 1 \cdot 3 - 1 \cdot 2 \\ &= 1 \cdot 3 - 1 \cdot (1 \cdot 5 - 1 \cdot 3). \end{aligned}$$

Vi förenklar uttrycket

$$\begin{aligned} 1 &= 1 \cdot 3 - 1 \cdot 2 \\ &= 1 \cdot 3 - 1 \cdot (1 \cdot 5 - 1 \cdot 3) \\ &= 1 \cdot 3 - 1 \cdot 5 + 1 \cdot 3 \\ &= \quad - 1 \cdot 5 + 2 \cdot 3. \end{aligned}$$

Slutligen skriver med hjälp av det första ledet i Euklides algoritmen ovan om 3:an som $3 = 1 \cdot 8 - 1 \cdot 5$ och förenklar sedan.

$$\begin{aligned} 1 &= 1 \cdot 3 - 1 \cdot 2 \\ &= 1 \cdot 3 - 1 \cdot (1 \cdot 5 - 1 \cdot 3) \\ &= 1 \cdot 3 - 1 \cdot 5 + 1 \cdot 3 \\ &= \quad - 1 \cdot 5 + 2 \cdot 3 \\ &= \quad - 1 \cdot 5 + 2 \cdot (1 \cdot 8 - 1 \cdot 5) \\ &= \quad - 1 \cdot 5 + 2 \cdot 8 - 2 \cdot 5 \\ &= \quad\quad\quad 2 \cdot 8 - 3 \cdot 5. \end{aligned}$$

Vi har alltså att $x = -3$ och $y = 2$ är en lösning till den diofantiska ekvationen $5x + 8y = 1$.

Exempel 1.8.2. Bestäm en lösning till diofantiska ekvationen $26x + 15y = 1$.

Lösning: Vi börjar med Euklides algoritm

$$\begin{aligned} 26 &= 1 \cdot 15 + 11 \\ 15 &= 1 \cdot 11 + 4 \\ 11 &= 2 \cdot 4 + 3 \\ 4 &= 1 \cdot 3 + 1. \end{aligned}$$

Sedan utför vi Euklides algoritm baklänges

$$\begin{aligned}
1 &= 1 \cdot 4 - 1 \cdot 3 \\
&= 1 \cdot 4 - 1 \cdot (1 \cdot 11 - 2 \cdot 4) \\
&= 1 \cdot 4 - 1 \cdot 11 + 2 \cdot 4 \\
&= \quad - 1 \cdot 11 + 3 \cdot 4 \\
&= \quad - 1 \cdot 11 + 3 \cdot (1 \cdot 15 - 1 \cdot 11) \\
&= \quad - 1 \cdot 11 + 3 \cdot 15 - 3 \cdot 11 \\
&= \qquad\qquad 3 \cdot 15 - 4 \cdot 11 \\
&= \qquad\qquad 3 \cdot 15 - 4 \cdot (1 \cdot 26 - 1 \cdot 15) \\
&= \qquad\qquad 3 \cdot 15 - 4 \cdot 26 + 4 \cdot 15 \\
&= \qquad\qquad\qquad - 4 \cdot 26 + 7 \cdot 15.
\end{aligned}$$

Vi får att en lösning till $26x + 15y = 1$ är: $x = -4$ och $y = 7$.

Övning 7. Bestäm en lösning till den diofantiska ekvationen

a) $47x + 16y = 1$

b) $12x + 17y = 1$

c) $39x + 14y = 1$

d) $187x + 29y = 1$.

I kommande exempel kommer vi att använda oss av modulobegreppet, som vi presenterar i ett senare avsnitt (1.10). Det enda man behöver veta just nu är att följande uttryck $ad \equiv 1 \pmod{m}$, där a, d och m är heltal, kan skrivas om till den diofantiska ekvationen $ad + my = 1$.

Exempel 1.8.3. Bestäm en lösning till d då $3d \equiv 1 \pmod{14}$.

Lösning: Vi kan skriva om uttrycket ovan som $3d + 14y = 1$. Denna diofantiska ekvation löser vi på samma sätt som vi gjort innan.

$$\begin{aligned}
14 &= 4 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1.
\end{aligned}$$

$$\begin{aligned}
1 &= 1 \cdot 3 - 1 \cdot 2 \\
&= 1 \cdot 3 - 1 \cdot (1 \cdot 14 - 4 \cdot 3) \\
&= 1 \cdot 3 - 1 \cdot 14 + 4 \cdot 3 \\
&= \quad - 1 \cdot 14 + 5 \cdot 3.
\end{aligned}$$

Vi får att $d = 5$ är en lösning.

Exempel 1.8.4. Beräkna d då $13d \equiv 1 \pmod{24}$ och $0 \leq d < 24$.

Lösning: Vi börjar med att beräkna $13d + 24y = 1$.

$$\begin{aligned} 24 &= 1 \cdot 13 + 11 \\ 13 &= 1 \cdot 11 + 2 \\ 11 &= 5 \cdot 2 + 1 \end{aligned}$$

$$\begin{aligned} 1 &= 1 \cdot 11 - 5 \cdot 2 \\ &= 1 \cdot 11 - 5 \cdot (1 \cdot 13 - 1 \cdot 11) \\ &= \quad - 5 \cdot 13 + 6 \cdot 11 \\ &= \quad - 5 \cdot 13 + 6 \cdot (1 \cdot 24 - 1 \cdot 13) \\ &= \quad\quad\quad 6 \cdot 24 - 11 \cdot 13. \end{aligned}$$

Vi får att $d = -11$ men det uppfyller inte $0 \leq d < 24$. Det korrekta värdet på d är något utav elementen i restklassen $[-11]_{24}$. Vi har

$$[-11]_{24} = \{\ldots, -35, -11, 13, 37, \ldots\}.$$

Vi ser att 13 är det enda element i restklassen ovan som uppfyller $0 \leq d < 24$. Alltså är $d = 13$.

Övning 8. Beräkna d då

a) $3d \equiv 1 \pmod{8}$ och $0 \leq d < 8$

b) $7d \equiv 1 \pmod{40}$ och $0 \leq d < 40$.

1.9 Binära tal

Vi är vana vid att räkna med decimala tal. Decimala tal har basen 10 och består av siffrorna $0 - 9$. Beroende på var en siffra befinner sig i ett tal har den olika värde. Den första siffran är värd mest och den sista är värd minst.

Exempel 1.9.1.

$$\begin{aligned} 6932 &= 6000 + 900 + 30 + 2 \\ &= 6 \cdot 10^3 + 9 \cdot 10^2 + 3 \cdot 10^1 + 2 \cdot 10^0. \end{aligned}$$

Binära tal har basen 2 och består av siffrorna 0 och 1. Beroende på var siffrorna befinner sig har de olika värde. Det första siffran är mest värd och

den sista är värd minst. Binära tal kan skrivas som decimala tal. Från och med nu kommer alla binära tal följas av en nedsänkt tvåa, t.ex. 1101_2.

Den sista siffran i ett binärt tal har värdet $2^0 = 1$.
Den näst sista siffran har värdet $2^1 = 2$.
Den tredje sista siffran har värdet $2^2 = 4$.
Den fjärde sista siffran har värdet $2^3 = 8$.
Den femte sista siffran har värdet $2^4 = 16$.
Den sjätte sista siffran har värdet $2^5 = 32$.

Det går att fortsätta på samma sätt men vi stannar här. Observera att talen hela tiden blir dubbelt så stora.

Exempel 1.9.2. Omvandla 1101_2 till ett decimalt tal.

Lösning: Vi börjar med den första (fjärde sista) siffran och ser att det är 1 och ger värdet $1 \cdot 8 = 8$. Sedan ser vi att den andra (tredje sista) siffran också är 1 och ger värdet $1 \cdot 4 = 4$. Den tredje (näst sista) siffran är 0 och ger värdet $0 \cdot 2 = 0$. Den fjärde (och sista) siffran är 1 och ger värdet $1 \cdot 1 = 1$. Slutligen lägger vi ihop allt

$$8 + 4 + 0 + 1 = 13.$$

Vi har alltså att $1101_2 = 13$.

Exempel 1.9.3. Skriv 101101_2 som ett decimalt tal.

Lösning:

$$\begin{aligned} 101101_2 &= 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= 1 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 \\ &= 45. \end{aligned}$$

Exempel 1.9.4. Vi har uppgifterna

a) Omvandla 19 till ett binärt tal.

b) Omvandla 52 till ett binärt tal.

Lösning:

a) Vi börjar med att undersöka vilka av talen $1, 2, 4, 8, 16, 32, ...$ vi måste addera för att få det decimala talet vi vill omvandla. Det finns en metod för att hitta vilka tal som ska adderas. Börja med det största av talen tidigare som är mindre än 19, vilket är 16. Nästa tal vi ska addera är det största av talen som är mindre än $19 - 16 = 3$, vilket är 2. Nästa tal vi adderar är det största av talen som är mindre än $3 - 2 = 1$, vilket är 1. Vi får då att $19 = 16 + 2 + 1$. Vi har alltså att

$$\begin{aligned} 19 &= 1 \cdot 16 + 0 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 \\ &= 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 \\ &= 10011_2. \end{aligned}$$

b)

$$\begin{aligned} 52 &= 1 \cdot 32 + 1 \cdot 16 + 0 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 0 \cdot 1 \\ &= 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 \\ &= 110100_2. \end{aligned}$$

Övning 9. Skriv talen på binär form

a) 9

b) 21

c) 68.

1.10 Modulusoperatorn och kongruenser

Heltalen 29 och 17 tillhör samma restklass om de divideras med 12. Detta eftersom vi har att

$$\begin{aligned} 29 &= 2 \cdot 12 + 5 \\ 17 &= 1 \cdot 12 + 5. \end{aligned}$$

De lämnar alltså båda resten 5. Heltalen 29 och 17 sägs därför vara *kungruenta modulo* 12. Vi skriver detta kortare som

$$29 \equiv 17 \pmod{12}.$$

Eller så kan vi skriva det som

$$29 \equiv_{12} 17.$$

Vi utläser båda ovanstående som att 29 är kongruent 17 modulo 12.

I kapitel 3 kommer vi, då a,b och m är heltal, att ha användning av ett annat sätt att avgöra om $a \equiv b \pmod{m}$. Detta sätt säger att om differensen $a - b$ är delbar med m så gäller det att $a \equiv b \pmod{m}$.

Exempel 1.10.1. Är $26 \equiv 12 \pmod{13}$?

Lösning: Vi har

$$26 = 2 \cdot 13 + 0$$
$$12 = 0 \cdot 13 + 12.$$

Svar: Nej, eftersom resterna är olika

Övning 10. Bestäm om

a) $13 \equiv 14 \pmod{3}$?

b) $57 \equiv_{14} 15$?

Exempel 1.10.2. Beräkna

a) $21 \pmod{6}$ (dvs vilken rest fås om 21 divideras med 6)

b) $53 \pmod{11}$.

Lösning: Vi har

a) $21 = 3 \cdot 6 + 3$
Svar: 3.

b) $53 = 4 \cdot 11 + 9$
Svar: 9.

Övning 11. Beräkna

a) $9 \pmod{3}$

b) $14 \pmod{6}$

c) $29 \pmod{33}$

d) $47 \pmod{13}$

e) $138 \pmod{46}$.

1.11 Moduloberäkningar med stora tal

Till att börja med vill vi påminna om två potenslagar som säger:

$$a^x \cdot a^y = a^{x+y}$$

$$(a^x)^y = a^{x \cdot y}.$$

Med hjälp av dessa ska vi presentera en metod, *binära metoden*, med vilken vi kan beräkna $a^x \pmod p$ snabbare än med den naiva metoden. Vi får användning av detta när vi inom RSA-kryptografin ska beräkna $a^x \pmod p$ för stora värden på x.

Exempel 1.11.1. Beräkna $7^9 \pmod{13}$.

Lösning: Vi börjar med att skriva 9 som om vi skulle omvandla 9 till ett binärt tal, vi kommer efter exemplet att förklara varför vi använder den binära metoden. Vi har alltså att

$$9 = 8 + 1.$$

Med hjälp av översta potenslagen ovan kan vi skriva $7^9 = 7^{8+1} = 7^8 \cdot 7^1$ Vi måste nu beräkna $7^8 \pmod{13}$ och $7^1 \pmod{13}$ och därefter multiplicera resultaten. Vi ser nedan att $7^1 \pmod{13}$ beräknas enkelt, men för att beräkna $7^8 \pmod{13}$ använder vi oss av den nedersta potenslagen ovan.

$$\begin{aligned}
7^1 &= 7 \equiv_{13} 7 \\
7^2 &= (7^1)^2 \equiv_{13} 7^2 = 49 \equiv_{13} 10 \\
7^4 &= (7^2)^2 \equiv_{13} 10^2 = 100 \equiv_{13} 9 \\
7^8 &= (7^4)^2 \equiv_{13} 9^2 = 81 \equiv_{13} 3.
\end{aligned}$$

Alltså är $7^8 \pmod{13} = 3$ och $7^1 \pmod{13} = 7$ och vi har att

$$7^9 = 7^8 \cdot 7^1 \equiv_{13} 3 \cdot 7 = 21 \equiv_{13} 8.$$

Alltså $7^9 \pmod{13} = 8$.

Vi ser att med den binära metoden krävdes fyra multiplikationer

$$\begin{aligned}
7^2 &= 49 \\
10^2 &= 100 \\
9^2 &= 81 \\
3 \cdot 7 &= 21
\end{aligned}$$

för att beräkna $7^9 \pmod{13} = 8$.

Om vi istället beräknar $7^9 \pmod{13} = 8$ med den naiva metoden får vi

$$\begin{aligned}
7^1 &= 7 \equiv_{13} 7 \\
7^2 &= (7^1) \cdot 7 \equiv_{13} 7 \cdot 7 = 49 \equiv_{13} 10 \\
7^3 &= (7^2) \cdot 7 \equiv_{13} 10 \cdot 7 = 70 \equiv_{13} 5 \\
7^4 &= (7^3) \cdot 7 \equiv_{13} 5 \cdot 7 = 35 \equiv_{13} 9 \\
7^5 &= (7^4) \cdot 7 \equiv_{13} 9 \cdot 7 = 63 \equiv_{13} 11 \\
7^6 &= (7^5) \cdot 7 \equiv_{13} 11 \cdot 7 = 77 \equiv_{13} 12 \\
7^7 &= (7^6) \cdot 7 \equiv_{13} 12 \cdot 7 = 84 \equiv_{13} 6 \\
7^8 &= (7^7) \cdot 7 \equiv_{13} 6 \cdot 7 = 42 \equiv_{13} 3 \\
7^9 &= (7^8) \cdot 7 \equiv_{13} 3 \cdot 7 = 21 \equiv_{13} 8.
\end{aligned}$$

Alltså åtta multiplikationer krävs för att beräkna $7^9 \pmod{13} = 8$. Det krävdes alltså fler multiplikationer med den naiva metoden än för den binära metoden, som i detta fall bara krävde fyra multiplikationer.

Det finns generella formler för att beräkna hur många multiplikationer de olika metoderna kräver. Om vi ska beräkna $x^e \pmod{n}$ så blir antalet multiplikationer med den naiva metoden

$$e - 1.$$

Med den binära metoden, där k är antalet siffror i e skrivet på binär form, blir det maximala antalet multiplikationer

$$2(k - 1).$$

Om vi nu ska beräkna $7^{1000000} \pmod{n}$ så blir antalet multiplikationer med den naiva metoden

$$e - 1 = 1000000 - 1 = 999999.$$

Eftersom 1000000 är 20 siffror långt på binär form blir det maximala antalet multiplikationer med den binära metoden

$$2(k - 1) = 2(20 - 1) = 38.$$

Vi ser alltså att om vi ska beräkna väldigt stora tal så kräver den binära metoden enormt mycket färre multiplikationer och är därför enormt mycket snabbare än den naiva metoden. Det är därför vi väljer att använda den binära metoden i denna bok.

Exempel 1.11.2. Beräkna $5^{12} \pmod{17}$.

Lösning: Vi har att

$$12 = 8 + 4$$

$$5^{12} = 5^{8+4} = 5^8 \cdot 5^4$$

$$\begin{aligned}
5^1 &= 5 \equiv_{17} 5 \\
5^2 &= (5^1)^2 \equiv_{17} 5^2 = 25 \equiv_{17} 8 \\
5^4 &= (5^2)^2 \equiv_{17} 8^2 = 64 \equiv_{17} 13 \\
5^8 &= (5^4)^2 \equiv_{17} 13^2 = 169 \equiv_{17} 16
\end{aligned}$$

$$5^{12} = 5^8 \cdot 5^4 \equiv_{17} 16 \cdot 13 = 208 \equiv_{17} 4.$$

Alltså är $5^{12} \pmod{17} = 4$.

Exempel 1.11.3. Beräkna $37^{23} \pmod{119}$.

Lösning: Vi löser detta på samma sätt som tidigare men vi skriver inte ut alla beräkningssteg lika noggrant.

$$\begin{aligned}
37^2 &= 1369 \equiv_{119} 60 \\
37^4 &\equiv_{119} 60^2 = 3600 \equiv_{119} 30 \\
37^8 &\equiv_{119} 30^2 = 900 \equiv_{119} 67 \\
37^{16} &\equiv_{119} 67^2 = 4489 \equiv_{119} 86 \\
37^{23} &= 37^{16} \cdot 37^4 \cdot 37^2 \cdot 37 \equiv_{119} 86 \cdot 30 \cdot 60 \cdot 37 = 5727600 \equiv_{119} 11.
\end{aligned}$$

Alltså är $37^{23} \pmod{119} = 11$.

Övning 12. Beräkna

a) $3^7 \pmod{11}$

b) $11^{13} \pmod{35}$

c) $29^{19} \pmod{77}$

d) $39^{43} \pmod{221}$.

Kapitel 2

RSA-kryptografi

2.1 Introduktion

Redan för flera tusen år sedan skickade människor hemliga meddelanden till varandra. Dessa meddelanden spelade stor roll i politiska maktspel och krig. Ett hemligt meddelande är ett krypto som i sin tur betyder att något döljer sin verkliga innebörd. När ett meddelande krypteras omvandlas det till ett krypto och vid dekryptering får man tillbaka det ursprungliga meddelandet. Speciellt matematiker har fascinerats av kryptografi och förmågan att skapa ett krypto som är så säkert som möjligt. För att ett krypto ska vara säkert måste dess krypteringsalgoritm vara svår att forcera, det vill säga knäcka. Krypteringsalgoritmen är en generell krypteringsformel där man med hjälp av en nyckel bestämmer hur den skall utföras. Nyckeln bestämmer alltså hur krypteringen skall genomföras på en text (meddelande).

Det som länge varit ett problem inom kryptografin var svårigheterna kring nyckeldistributionen. När två personer, vi kallar dem för sändare och mottagare, skall utväxla ett hemligt meddelande i form av ett krypto, måste de ha kommit överens om en nyckel. Detta för att kryptering och dekryptering skall vara möjlig. Nyckeln måste vara hemlig, vilket blir problematiskt eftersom för att kunna utväxla en hemlighet (meddelandet) måste personerna först redan ha kommit överens om en annan hemlighet (nyckeln). Problemet fick en lösning 1977 då Rivest, Shamir och Adleman presenterade kryptosystemet RSA. Det är en kryptering med öppen (offentlig) nyckel. Krypteringsalgoritmen är utformad på så sätt att mottagaren har en offentlig nyckel som sändaren letar upp och krypterar sitt meddelande med. Sedan använder mottagaren en privat nyckel (hemlig) för att dekryptera meddelandet.

Inom RSA-kryptografin används många beteckningar som är nödvändiga att känna till för att kunna utföra RSA-kryptografi. Dessa beteckningar skiljer sig lite i olika böcker och artiklar men vi har valt de beteckningar som vi upplevt varit mest vanliga och lätta att förstå. Nedan förklaras beteckningarna.

$$
\begin{aligned}
p &= \text{ett primtal} \\
q &= \text{ett annat primtal} \\
n &= p \cdot q \\
m &= (p-1)(q-1) \\
e &= \text{ett heltal som uppfyller sgd}(e,m) = 1 \text{ och } 1 < e < m \\
d &= \text{ett heltal som uppfyller } ed \equiv 1 \pmod{m} \text{ och } 1 < d < m \\
x &= \text{ett hemligt meddelande} \\
y &= \text{det nya meddelandet som fås då } x \text{ krypteras} \\
E(x) &= x^e \pmod{n} \text{ är funktionen som krypterar } x \text{ till } y \\
D(y) &= y^d \pmod{n} \text{ är funktionen som dekrypterar } y \text{ till } x.
\end{aligned}
$$

I föregående kapitel när vi räknade med restklasser var vi intresserade av elementet d som uppfyllde villkoret $0 \leq d < m$. Men inom RSA-kryptografin används istället villkoret $1 < d < m$. Detta eftersom att $d = 0$ inte kan uppfylla $ed \equiv 1 \pmod{m}$ och om $d = 1$ måste $e = 1$, vilket blir trivialt. RSA-kryptografin är därför inte användbar om $d = 0$ eller $d = 1$.

I nedanstående exempel kommer vi förklara de generella principerna för hur ovanstående beteckningar används inom RSA-kryptografin.

Exempel 2.1.1. Bert väljer p och q och beräknar n och m. Han väljer sedan ett e och räknar ut d. Bert offentliggör n och e så att alla vet vad de är. Men d, p, q och m håller Bert hemliga. Krypteringsfunktionen $E(x)$ och dekrypteringsfunktionen $D(y)$ är alltid offentliga, det vill säga att alla alltid har tillgång till dem.

Anna vill skicka ett hemligt meddelande x till Bert utan att någon annan kan ta reda på x. Eftersom Anna känner till x, e och n använder hon $E(x)$ för att räkna fram y. Anna skickar y till Bert.

När y skickas till Bert lyckas Calle se vad y är. Men som vi kommer att se i kommande avsnitt räcker det inte med att veta y, n och e för att kunna dekryptera y till x. Calle behöver nämligen något utav p, q eller m för att kunna räkna ut d, vilket krävs för att kunna dekryptera y till x. Calle kan alltså inte räkna ut x.

När Bert får det krypterade meddelandet y dekrypterar han det med hjälp av $D(y)$ till Annas ursprungliga meddelande x.

Listan nedan är en sammanfattning av ovanstående text.

Bert vet: p, q, n, m, e och d
Anna vet: n, e och x
Anna beräknar: $E(x) = x^e \pmod{n} = y$
Anna skickar: y till Bert
Bert tar emot: y
Bert beräknar: $D(y) = y^d \pmod{n} = x$
Calle vet: n och e
Calle snokar fram: y
Calle behöver: d, p, q eller m för att beräkna x
Calle kan inte räkna fram: d, p, q eller m
Calle kan inte räkna fram: x.

I avsnitt $2.2 - 2.5$ kommer alla meddelanden x bestå av små tal. Men givetvis måste ett meddelande kunna bestå av mer än ett litet tal. Vi kommer därför i avsnitt 2.6 att förklara principerna för hur ett skriftligt meddelande kan krypteras med RSA-kryptografi.

2.2 Skapa offentlig nyckel och beräkna d

Exempel 2.2.1. Vi ska nu visa ett exempel på hur vi räknar fram p, q, n, m, e och d. Till att börja med väljer vi två primtal p och q. För att RSA-kryptografin ska vara säker måste p och q vara primtal som innehåller minst 100 siffror, men vi återkommer till detta senare och väljer istället de små primtalen $p = 3$ och $q = 23$. Vi beräknar n och m

$$n = p \cdot q = 3 \cdot 23 = 69$$

$$m = (p-1)(q-1) = 2 \cdot 22 = 44.$$

Vi ska sedan välja ett e som uppfyller sgd(e,m) = 1 och $1 < e < m$. Vi väljer $e = 7$ eftersom sgd(7,44) = 1 och $1 < 7 < 44$. Nästa moment är att beräkna d som ska uppfylla $ed \equiv 1 \pmod{m}$ och $1 < d < m$. Vi har

$$7d \equiv 1 \pmod{44}.$$

Genom att lösa den diofantiska ekvationen $7d+44y = 1$ får vi fram ett värde på d.

$$\begin{aligned} 44 &= 6 \cdot 7 + 2 \\ 7 &= 3 \cdot 2 + 1 \end{aligned}$$

$$\begin{aligned} 1 &= 1 \cdot 7 - 3 \cdot 2 \\ &= 1 \cdot 7 - 3 \cdot (1 \cdot 44 - 6 \cdot 7) \\ &= \quad -3 \cdot 44 + 19 \cdot 7. \end{aligned}$$

Vi får $d = 19$. Vi har nu allt som behövs för att kunna dekryptera krypterade meddelanden. Vi offentliggör n och e så att vem som helst ska kunna kryptera sitt meddelande och skicka det till oss så att vi kan dekryptera det krypterade meddelandet och därigenom se det ursprungliga meddelandet.

Exempel 2.2.2. Vi väljer $p = 5$ och $q = 13$ och får att

$$n = 5 \cdot 13 = 65$$

$$m = 4 \cdot 12 = 48.$$

Vi väljer $e = 5$ eftersom sgd(5,48) $= 1$ och $1 < 5 < 48$. Nu ska vi bestämma d genom att beräkna kongruensen $5d \equiv 1 \pmod{48}$. Vi har

$$\begin{aligned} 48 &= 9 \cdot 5 + 3 \\ 5 &= 1 \cdot 3 + 2 \\ 3 &= 1 \cdot 2 + 1 \end{aligned}$$

$$\begin{aligned} 1 &= 1 \cdot 3 - 1 \cdot 2 \\ &= 1 \cdot 3 - 1 \cdot (1 \cdot 5 - 1 \cdot 3) \\ &= \quad -1 \cdot 5 + 2 \cdot 3 \\ &= \quad -1 \cdot 5 + 2 \cdot (1 \cdot 48 - 9 \cdot 5) \\ &= \quad\quad\quad 2 \cdot 48 - 19 \cdot 5. \end{aligned}$$

Vi får att $d = -19$ men det uppfyller inte $1 < d < 48$. Det korrekta värdet på d är något utav elementen i restklassen $[-19]_{48}$. Vi har

$$[-19]_{48} = \{\ldots, -19, 29, 77 \ldots\}.$$

Vi ser att 29 är det enda element i restklassen ovan som uppfyller $1 < d < 48$. Alltså är $d = 29$.

Övning 13. Beräkna n, m och d om

a) $p = 7$, $q = 13$ och $e = 7$

b) $p = 11$, $q = 17$ och $e = 11$.

2.3 Kryptering

Exempel 2.3.1. Vi ska nu kryptera meddelandet $x = 11$. Vi hämtar $n = 69$ och $e = 7$ från Exempel 2.2.1. För att kryptera måste vi använda krypteringsfunktionen $E(x) = x^e \pmod{n} = y$. Vi sätter in värdena för x, n och e i funktionen och får att

$$E(11) = 11^7 \pmod{69} = y$$

$$11^2 = 121 \equiv_{69} 52$$
$$11^4 \equiv_{69} 52^2 = 2704 \equiv_{69} 13$$
$$11^7 = 11^4 \cdot 11^2 \cdot 11 \equiv_{69} 13 \cdot 52 \cdot 11 = 7436 \equiv_{69} 53.$$

Vi har nu krypterat meddelandet $x = 11$ till det krypterade meddelandet $y = 53$.

Exempel 2.3.2. Vi ska nu kryptera meddelandet $x = 23$. Vi hämtar $n = 65$ och $e = 5$ från Exempel 2.2.2. För att kryptera måste vi använda krypteringsfunktionen $E(x) = x^e \pmod{n} = y$. Vi sätter in värdena för x, n och e i funktionen och får att

$$E(23) = 23^5 \pmod{65} = y$$

$$23^2 = 529 \equiv_{65} 9$$
$$23^4 \equiv_{65} 9^2 = 81 \equiv_{65} 16$$
$$23^5 = 23^4 \cdot 23 \equiv_{65} 16 \cdot 23 = 368 \equiv_{65} 43.$$

Vi har nu krypterat meddelandet $x = 23$ till det krypterade meddelandet $y = 43$.

Övning 14. Kryptera x då

a) $x = 5$, $e = 7$ och $n = 91$ (e och n är hämtade från övning 13 a)

b) $x = 7$, $e = 11$ och $n = 187$ (e och n är hämtade från övning 13 b).

2.4 Dekryptering

Exempel 2.4.1. Vi ska nu dekryptera meddelandet $y = 53$ som vi får från Exempel 2.3.1. Vi hämtar $n = 69$ och $d = 19$ från Exempel 2.2.1. För att dekryptera måste vi använda dekrypteringsfunktionen $D(y) = y^d \pmod{n} = x$. Vi sätter in värdena för y, n och d i funktionen och får att

$$D(53) = 53^{19} \pmod{69} = x$$

$$
\begin{aligned}
53^2 &= 2809 \equiv_{69} 49 \\
53^4 &\equiv_{69} 49^2 = 2401 \equiv_{69} 55 \\
53^8 &\equiv_{69} 55^2 = 3025 \equiv_{69} 58 \\
53^{16} &\equiv_{69} 58^2 = 3364 \equiv_{69} 52 \\
53^{19} &= 53^{16} \cdot 53^2 \cdot 53 \equiv_{69} 52 \cdot 49 \cdot 53 = 135044 \equiv_{69} 11.
\end{aligned}
$$

Vi har nu dekrypterat meddelandet $y = 53$ till det ursprungliga meddelandet $x = 11$ som vi hade i Exempel 2.3.1.

Exempel 2.4.2. Vi ska nu dekryptera meddelandet $y = 43$ som vi får från Exempel 2.3.2. Vi hämtar $n = 65$ och $d = 29$ från Exempel 2.2.2. För att dekryptera måste vi använda dekrypteringsfunktionen $D(y) = y^d \pmod{n} = x$. Vi sätter in värdena för y, n och d i funktionen och får att

$$D(43) = 43^{29} \pmod{65} = x$$

$$
\begin{aligned}
43^2 &= 1849 \equiv_{65} 29 \\
43^4 &\equiv_{65} 29^2 = 841 \equiv_{65} 61 \\
43^8 &\equiv_{65} 61^2 = 3721 \equiv_{65} 16 \\
43^{16} &\equiv_{65} 16^2 = 256 \equiv_{65} 61 \\
43^{29} &= 43^{16} \cdot 43^8 \cdot 43^4 \cdot 43 \equiv_{65} 61 \cdot 16 \cdot 61 \cdot 43 = 2560048 \equiv_{65} 23.
\end{aligned}
$$

Vi har nu dekrypterat meddelandet $y = 43$ till det ursprungliga meddelandet $x = 23$ som vi hade i Exempel 2.3.2.

Övning 15. Dekryptera y då

a) $y = 47$, $d = 31$ och $n = 91$ (d och n är hämtade från övning 13 a och y är hämtat från övning 14 a)

b) $y = 150$, $d = 131$ och $n = 187$ (d och n är hämtade från övning 13 b och y är hämtat från övning 14 b).

2.5 Skapa offentlig nyckel, bestämma d, kryptera och dekryptera

Vi kommer i exemplet nedan att utföra alla momenten i RSA-kryptografi med små tal. Samtliga moment i exemplet har utförts tidigare i boken men nu kommer allt göras i ett enda exempel och allt kommer inte förklaras lika noggrant som tidigare.

Exempel 2.5.1. Inledningsvis väljer vi $p = 5$ och $q = 11$ och får att

$$n = 5 \cdot 11 = 55$$

$$m = 4 \cdot 10 = 40.$$

Vi väljer $e = 7$ eftersom sgd(7,40) $= 1$ och $1 < 7 < 40$. Nu ska vi bestämma d genom att beräkna kongruensen $7d \equiv 1 \pmod{40}$. Vi har

$$\begin{aligned} 40 &= 5 \cdot 7 + 5 \\ 7 &= 1 \cdot 5 + 2 \\ 5 &= 2 \cdot 2 + 1 \end{aligned}$$

$$\begin{aligned} 1 &= 1 \cdot 5 - 2 \cdot 2 \\ &= 1 \cdot 5 - 2 \cdot (1 \cdot 7 - 1 \cdot 5) \\ &= \quad - 2 \cdot 7 + 3 \cdot 5 \\ &= \quad - 2 \cdot 7 + 3 \cdot (1 \cdot 40 - 5 \cdot 7) \\ &= \quad\quad\quad 3 \cdot 40 - 17 \cdot 7. \end{aligned}$$

Vi får att $d = -17$ men det uppfyller inte $1 < d < 40$. Det korrekta värdet på d är något utav elementen i restklassen $[-17]_{40}$. Vi har

$$[-17]_{40} = \{\ldots, -17, 23, 63 \ldots\}.$$

Vi ser att 23 är det enda element i restklassen ovan som uppfyller $1 < d < 40$. Alltså är $d = 23$.

Vi ska nu kryptera meddelandet $x = 8$ och använder krypteringsfunktionen $E(x) = x^e \pmod{n} = y$. Vi sätter in värdena för x, n och e i funktionen och får att

$$E(8) = 8^7 \pmod{55} = y$$

$$\begin{aligned} 8^2 &= 64 \equiv_{55} 9 \\ 8^4 &\equiv_{55} 9^2 = 81 \equiv_{55} 26 \\ 8^7 &= 8^4 \cdot 8^2 \cdot 8 \equiv_{55} 26 \cdot 9 \cdot 8 = 1872 \equiv_{55} 2. \end{aligned}$$

Vi har nu krypterat meddelandet $x = 8$ till det krypterade meddelandet $y = 2$.

Vi ska nu dekryptera meddelandet $y = 2$. Vi har $n = 55$ och $d = 23$. För att dekryptera måste vi använda dekrypteringsfunktionen $D(y) = y^d \pmod{n} = x$. Vi sätter in värdena för y, n och d i funktionen och får att

$$D(2) = 2^{23} \pmod{55} = x$$

$$
\begin{aligned}
2^2 &= 4 \equiv_{55} 4 \\
2^4 &\equiv_{55} 4^2 = 16 \equiv_{55} 16 \\
2^8 &\equiv_{55} 16^2 = 256 \equiv_{55} 36 \\
2^{16} &\equiv_{55} 36^2 = 1296 \equiv_{55} 31 \\
2^{23} &= 2^{16} \cdot 2^4 \cdot 2^2 \cdot 2 \equiv_{55} 31 \cdot 16 \cdot 4 \cdot 2 = 3968 \equiv_{55} 8.
\end{aligned}
$$

Vi har nu dekrypterat meddelandet $y = 2$ till det ursprungliga meddelandet $x = 8$.

Övning 16. Beräkna n, m, d samt kryptera x till y. Dekryptera därefter tillbaka y till x. Informationen du har tillgänglig är att $p = 7$, $q = 11$, $e = 17$ och $x = 6$.

Övning 17. Genomför på egen hand alla stegen i RSA-kryptografi. Alltså ska p, q, n, m, e, d, x och y bestämmas. Något som inte nämnts tidigare är att x måste vara mindre än n. För att det inte ska bli allt för jobbiga uträkningar rekommenderas att $2 < p, q, x < 15$ och som nämndes tidigare måste $x < n$.

Övning 18. Utför RSA-kryptografi tillsammans med en kompis som kan RSA-kryptografi. Skapa krypteringsnyckel och dekrypteringstalet. Skicka krypteringsnyckeln till kompisen men behåll krypteringstalet hemligt. Be kompisen kryptera ett meddelande ($5 < x < 20$ samt $x < n$) och skicka det krypterade meddelandet till dig. Du dekrypterar därefter det krypterade meddelandet och får fram kompisens ursprunliga meddelande.

2.6 RSA-kryptografi med skriftliga meddelanden

I detta avsnitt kommer några grundläggande principer angående RSA-kryptografi med skriftliga meddelanden att beskrivas. Inga fullständiga exempel kommer beskrivas eftersom det blir väldigt stora tal som blir jobbiga att beräkna.

Exempel 2.6.1. Antag att vi vet att $n = 3127$ och att vi vill kryptera det hemliga meddelandet "ÅK HEM". Vi börjar med att ge ett tal till varje bokstav i alfabetet samt till mellanslag. Vi har: Mellanslag = 00, A = 01,

B = 02, C = 03, och så vidare. Bokstäverna i "ÅK HEM" blir då:

$$\begin{aligned} \text{Å} &= 27 \\ \text{K} &= 11 \\ \text{Mellanslag} &= 00 \\ \text{H} &= 08 \\ \text{E} &= 05 \\ \text{M} &= 13. \end{aligned}$$

Nu skulle vi kunna skriva "ÅK HEM" som 271100080513 och sätta detta som x och kryptera det. Men det fungerar inte eftersom $n = 3127$ och vi från övning 17 fick veta att RSA-kryptografi kräver att $x < n$. Det vi istället kan göra är att dela upp meddelandet i mindre delar. Frågan är hur små bitarna måste vara. Om vi väljer två bokstäver (eller en bokstav och ett mellanslag) som uppdelning av meddelandet kan x bli maximalt 2929, det vill säga bokstavskombinationen "ÖÖ". Väljer vi tre bokstäver får vi ett maximalt värde på x till 292929, som motsvarar "ÖÖÖ". Eftersom

$$2929 < n = 3127 < 292929$$

och vi vill att $x < n$ så väljer vi att dela upp meddelandet i delar med två bokstäver (eller en bosktav och mellanslag) i varje del. "ÅK HEM" delas därför upp i

$$\begin{aligned} \text{“ÅK”} &= 2711 = x_1 \\ \text{“ H”} &= 0008 = x_2 \\ \text{“EM”} &= 0513 = x_3. \end{aligned}$$

Observera att det är ett mellanslag innan H. Vi kan nu kryptera de tre talen x_1, x_2 och x_3 var för sig genom att använda krypteringsalgoritmen och få

$$\begin{aligned} E(x_1) &= {x_1}^e \pmod n = y_1 \\ E(x_2) &= {x_2}^e \pmod n = y_2 \\ E(x_3) &= {x_3}^e \pmod n = y_3. \end{aligned}$$

Vi skickar sedan kryptotalen, det vill säga de tre nya tal y_1, y_2 och y_3 som erhållts genom krypteringen. Mottagaren kan sedan dekryptera dessa kryptotal y_1, y_2 och y_3 med hjälp av dekrypteringssalgoritmen och få

$$\begin{aligned} D(y_1) &= {y_1}^d \pmod n = x_1 = 2711 \\ D(y_2) &= {y_2}^d \pmod n = x_2 = 0008 \\ D(y_3) &= {y_3}^d \pmod n = x_3 = 0513. \end{aligned}$$

Mottagaren sätter ihop talen till 271100080513 och översätter det enkelt till meddelandet “ÅK HEM”.

Detta exempel var förenklat med tanke på att vi inte räknade ut m, e, d, y_1, y_2 och y_3. För att se ett komplett exempel där alla moment blir uträknade rekommenderar vi vidare läsning i *RSA-kryptering i enkla steg*.

I praktiken skickas självklart längre och mer komplicerade meddelanden än i exemplet ovan, som dessutom är väldigt enkelt att forcera. Ju fler kryptotal som blir tillgängliga desto enklare blir det att se samband och slutligen forcera kryptona. Exempelvis skulle det efter ett tag bli tydligt att kombinationen 19 förekommer klart oftare är 26, eftersom S = 19 är en mer frekvent bokstav än Z = 26. Det skulle också märkas att inga tal överstiger cirka 2929 = “ÖÖ” och att kombinationen av de två sista siffrorna aldrig överstiger 29 = “Ö”. All denna information gör det enkelt att forcera. För att komma åt dessa problem finns flera avancerade metoder som försvårar forceringen. Det kan exempelvis handla om att överflödiga kombinationer elimineras innan krypteringen eller att identiska meddelandeblock krypteras på olika sätt.

Kapitel 3

Matematiken bakom RSA-kryptografin

3.1 Begreppet $\varphi(m)$

Om m är ett positivt heltal, så är $\varphi(m)$ antalet positiva heltal mindre än eller lika med m som dessutom är relativt prima med m.

Exempel 3.1.1. Bestäm $\varphi(8)$.

Lösning: Vi får att 1,3,5,7 är relativt prima med, och dessutom mindre än, 8. Eftersom vi har 4 stycken tal får vi att $\varphi(m) = 4$.

Det finns vissa räkneregler som förenklar beräkningar av $\varphi(m)$. För alla primtal p, och alla heltal a, b gäller följande

$$\begin{aligned}\varphi(p) &= p - 1\\ \varphi(p^n) &= p^n(1 - 1/p)\\ \varphi(ab) &= \varphi(a)\varphi(b).\end{aligned}$$

Exempel 3.1.2. Bestäm $\varphi(13)$ och $\varphi(28)$.

Lösning: Eftersom 13 är ett primtal kan vi använda oss av den första räkneregeln som ger

$$\varphi(13) = 13 - 1 = 12.$$

Talet 28 är inget primtal, men vi vet från kapitel 1.5 att varje sammansatt tal kan skrivas som en produkt av primtal. Räkneregel 3 gör det möjligt för oss att faktorisera 28 i primtalsfaktorer

$$28 = 4 \cdot 7 = 2 \cdot 2 \cdot 7 = 2^2 \cdot 7.$$

Vi kan nu använda oss av räkneregel 1, 2 och 3

$$\begin{aligned}\varphi(7) &= 6\\ \varphi(2^2) &= 2^2(1-1/2) = 2\\ \varphi(28) &= \varphi(7)\varphi(2^2) = 6 \cdot 2 = 12.\end{aligned}$$

Övning 19. Bestäm $\varphi(40)$.

3.2 Prima restmängd modulo m

Exempel 3.2.1. Vi skall nu undersöka vilka rester vi får vid modulo 8. De minsta positiva resterna är

$$\{1,2,3,4,5,6,7\}\,.$$

Nu undersöker vi vilka tal i mängden som är relativt prima med 8 och bildar med dessa tal den nya mängden

$$\{1,3,5,7\}\,.$$

Talen i ovanstående mängd lämnar alla olika rest vid modulo 8. Vi säger att de är *parvis inkongruenta modulo* 8. En annan mängd där elementen har olika rest vid modulo 8 är

$$\{3,9,15,21\}\,.$$

Elementen är även i denna mängd realtivt prima med 8 och dessutom parvis inkongruenta modulo 8. Mängderna $\{1,3,5,7\}$ och $\{3,9,15,21\}$ kallar vi för *prima restmängder modulo* 8. I *Talteori och kryptografi* beskriver man dessa mängder som *reducerade mängder rester modulo* 8.

Definition 3.2.1. En *prima restmängd modulo* m är en mängd som uppfyller samtliga följande villkor:

- Mängden består av exakt $\varphi(m)$ stycken element.
- Varje element i mängden är relativt prima med m.
- Elementen är parvis inkongruenta modulo m.

Exempel 3.2.2. Bestäm en prima restmängd modulo 10.

Lösning: Vi tar reda på $\varphi(10)$

$$\begin{aligned}\varphi(10) &= \varphi(5)\varphi(2)\\ &= (5-1)\cdot(2-1)\\ &= 4.\end{aligned}$$

En prima restmängd modulo 10 skall alltså bestå av 4 element. Mängden av de minsta positiva resterna modulo 10 är

$$\{1,2,3,4,5,6,7,8,9\}\,.$$

Vi tar reda på vilka tal i denna mängd som är relativt prima med 10. Både 2 och 5 delar 10, så de är inte relativt prima med 10. Detta leder till att även 4, 6 och 8 inte är relativt prima med 10 eftersom de alla är delbara med 2. Vi har att sgd(10,1) = 1 är givet, eftersom 1 är relativt primt med alla heltal. Vi undersöker de resterande talen 3, 7 och 9. Vi har

$$\begin{aligned}10 &= 5\cdot 2\\ 3 &= 3\cdot 1\\ 7 &= 7\cdot 1\\ 9 &= 3\cdot 3.\end{aligned}$$

Alltså är sgd(10,1) = 1, sgd(10,3) = 1, sgd(10,7) = 1 och sgd(10,9) = 1. Därmed har vi fått fram de tal som uppfyller det andra villkoret för en prima restmängd modulo 10. Det återstår att undersöka om dessa fyra tal är parvis inkongruenta modulo 10.

$$\begin{aligned}1 &= 0\cdot 10+1\\ 3 &= 0\cdot 10+3\\ 7 &= 0\cdot 10+7\\ 9 &= 0\cdot 10+9.\end{aligned}$$

Vi kan konstatera att alla talen lämnar olika rest vid modulo 10 och de är parvis inkongruenta. Mängden $\{1,3,7,9\}$ är alltså en prima restmängd modulo 10.

Övning 20. Bestäm en prima restmängd modulo 14.

3.3 Prima restmängd och restklasser

Exempel 3.3.1. Låt oss titta närmare på en prima restmängd modulo 12. Vi börjar att ta reda på $\varphi(12)$

$$\begin{aligned}\varphi(12) &= \varphi(3)\varphi(2^2)\\ &= (3-1)\cdot 2^2(1-1/2)\\ &= 2\cdot 2\\ &= 4.\end{aligned}$$

En prima restmängd modulo 12 består alltså av 4 element. Vi kallar denna mängd för

$$\{r_1,r_2,r_3,r_4\}\,.$$

Vi undersöker vilka tal som är relativt prima med 12. Det kommer att finnas oändligt många, eftersom det finns oändligt många primtal. Vi avgränsar oss därför och undersöker alla positiva heltal till och med 40. Vi får att

$$\begin{aligned}&1,5,7,11,13,17,19,\\ &23,25,29,31,35,37\end{aligned}$$

är relativt prima med 12. Vid en närmare undersökning kan vi dela upp talen i fyra talföljder som alla ökar med 12.

$$\begin{aligned}&1,13,25,37\\ &5,17,29\\ &7,19,31\\ &11,23,35.\end{aligned}$$

De fyra talföljderna ingår i olika restklasser modulo 12.

$$\begin{aligned}[1]_{12} &= \{\ldots,-11,1,13,25,37,\ldots\}\\ [5]_{12} &= \{\ldots,-7,5,17,29,41,\ldots\}\\ [7]_{12} &= \{\ldots,-5,7,19,31,43,\ldots\}\\ [11]_{12} &= \{\ldots,-1,11,23,35,47,\ldots\}.\end{aligned}$$

Alla talen i en restklass är kongruenta med varandra, men inga tal från olika restklasser modulo 12 är kongruenta med varandra. För att bilda en prima restmängd modulo 12 väljer vi ut ett tal från varje restklass. Vi väljer

$$\begin{aligned}r_1 &= 13 \text{ från } [1]_{12}\\ r_2 &= 17 \text{ från } [5]_{12}\\ r_3 &= 43 \text{ från } [7]_{12}\\ r_4 &= 47 \text{ från } [11]_{12}.\end{aligned}$$

Vi får alltså att {13,17,43,47} är en prima restmängd modulo 12. Detta eftersom att alla element är relativt prima med 12 och parvis inkongruenta modulo 12.

3.4 Bevis för prima restmängd

Sats 3.4.1. *Om vi har en prima restmängd modulo* m

$$\left\{r_1,r_2,...,r_{\varphi(m)}\right\}$$

och ett heltal a *som uppfyller* $\text{sgd}(a,m) = 1$*, så är också*

$$\left\{ar_1,ar_2,...,ar_{\varphi(m)}\right\}$$

en prima restmängd modulo m*.*

Bevis. Vi delar upp beviset i två delar. Eftersom en prima restmängd modulo m måste uppfylla att $\text{sgd}(ar_j,m) = 1$ och att inga element från $\left\{ar_1,ar_2,...,ar_{\varphi(m)}\right\}$ är parvis kongruenta modulo m. Vi börjar med det första villkoret och antar att $\text{sgd}(ar_j,m) > 1$ där r_j kan vara vilket element som helst i den prima restmängden modulo m. Vi vet då att det finns ett primtal p sådant att

$$p|\text{sgd}(ar_j,m) > 1 \Rightarrow \ p|a \text{ och } p|m \text{ eller } p|r_j \text{ och } p|m.$$

Eftersom att a och m är relativt prima, kan inte p dela både a och m, på grund av att att $\text{sgd}(a,m) = 1$. Vi har även att r_j är något element i den prima restmängden modulo m. Vilket innebär att $\text{sgd}(r_j,m) = 1$, så p kan inte heller dela r_j och m. Alltså kan inte $\text{sgd}(ar_j,m) > 1$ vara sant, vilket leder till att $\text{sgd}(ar_j,m) = 1$.

För att bevisa det andra villkoret antar vi att

$$ar_j \equiv ar_k \pmod{m} \text{ för } j \neq k.$$

Vi skriver om det som

$$m|(ar_j - ar_k) \Leftrightarrow m|a(r_j - r_k).$$

Eftersom $\text{sgd}(a,m) = 1$ så kan inte m dela a och vi får

$$m|(r_j - r_k) \Leftrightarrow r_j \equiv r_k \pmod{m}.$$

Detta motsäger dock att

$$\{r_1,r_2,...,r_{\varphi(m)}\}$$

är en prima restmängd modulo m. Detta innebär att ar_j och ar_k inte kan vara kongruenta modulo m, vilket är det vi från början ville bevisa. □

Sats 3.4.2. *För positiva heltal m och a sådana att* $\text{sgd}(a,m) = 1$*, så gäller* $a^{\varphi(m)} \equiv 1 \pmod{m}$.

Bevis. Vi låter

$$\{r_1,r_2,...,r_{\varphi(m)}\}$$

vara den minsta positiva mängden av den prima restmängden modulo m, vilket innebär att alla element måste vara mindre än m, dvs $0 < r_j < m$ för alla j. Vi vet från tidigare bevis att $\text{sgd}(a,m) = 1$, ger att

$$\{ar_1,ar_2,...,ar_{\varphi(m)}\}$$

också är en prima restmängd modulo m. Alla tal a som är relativt prima med m ingår i olika restklasser modulo m. Elementen $r_1, r_2, ...,r_{\varphi(m)}$ ingår i varsin sådan restklass (se exempel 3.3.1). Detta kommer innebära att elementen $ar_1, ar_2,...,ar_{\varphi(m)}$ också kommer ingå i varsin restklass modulo m. Eftersom $\{ar_1,ar_2,...,ar_{\varphi(m)}\}$ är en prima restmängd är alla element parvist inkongruenta, vilket betyder att vart och ett av elementen $ar_1, ar_2,...,ar_{\varphi(m)}$ kommer ingå i samma restklass som något av elementen $r_1, r_2,...,r_{\varphi(m)}$. Alltså måste för varje j gälla att

$$ar_j \equiv r_k \pmod{m}$$

för exakt ett k. Vi skriver om ovanstående som

$$\begin{aligned} & ar_1 \cdot ar_2 \cdot ... \cdot ar_{\varphi(m)} \equiv r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)} \pmod{m} \\ \Rightarrow & a^{\varphi(m)} \cdot r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)} \equiv r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)} \pmod{m} \\ \Rightarrow & a^{\varphi(m)} \equiv 1 \pmod{m}. \end{aligned}$$

Det sista steget förklaras av att

$$\begin{aligned} & a^{\varphi(m)} \cdot r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)} \equiv r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)} \pmod{m} \Leftrightarrow \\ & m|(a^{\varphi(m)} \cdot (r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)}) - (r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)})) \Leftrightarrow \\ & m|(a^{\varphi(m)} - 1) \cdot (r_1 \cdot r_2 \cdot ... \cdot r_{\varphi(m)}). \end{aligned}$$

Eftersom samtliga r_j är relativt prima med m, så måste $m|(a^{\varphi(m)} - 1)$ och vi får att $a^{\varphi(m)} \equiv 1 \pmod{m}$. □

Sats 3.4.3. *Om p är ett primtal och a ett heltal, så gäller $a^{p-1} \equiv 1 \pmod{p}$, förutsatt att p inte delar a.*

Bevis. Eftersom $\varphi(p) = p - 1$ och $a^{\varphi(m)} \equiv 1 \pmod{m}$ har vi att $a^{p-1} \equiv 1 \pmod{p}$. □

3.5 Varför dekryptering alltid ger tillbaka x

Vi vet att när vi krypterar så använder vi funktionen

$$E(x) = x^e \pmod{n} = y$$

och när vi dekrypterar använder vi funktionen

$$D(y) = y^d \pmod{n} = x.$$

Sats 3.5.1. *Om kryptering och dekryptering skrivs som en funktion fås*

$$D(E(x)) = x.$$

Bevis. Ekvationen ovan säger att om $ed \equiv 1 \pmod{m}$ och $0 < x < n - 1$ har vi att

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

Vi ska nu bevisa att det verkligen stämmer. Vi börjar med att visa $x^{ed} \equiv x \pmod{p}$. Om primtalet p delar x så delar p även x^{ed}. Då får vi $x \equiv 0 \pmod{p}$ och $x^{ed} \equiv 0 \pmod{p}$ och detta ger att $x^{ed} \equiv x \pmod{p}$.

Vi ska nu visa att $x^{ed} \equiv x \pmod{p}$ även när p inte delar x. Vi vet sedan tidigare att $m = (p-1)(q-1)$ och att $ed \equiv 1 \pmod{m}$. Då kan vi skriva att $ed = 1 + km = 1 + k(p-1)(q-1)$, där k är ett heltal. Vi har

$$x^{ed} = x^{1+k(p-1)(q-1)} = x \cdot x^{k(p-1)(q-1)} = x(x^{p-1})^{k(q-1)}$$

och med hjälp av Fermats lilla sats (se sats 3.4.3) som säger att $x^{p-1} \equiv 1 \pmod{p}$ fortsätter vi ovanstående

$$x(x^{p-1})^{k(q-1)} \equiv_p x \cdot 1^{k(q-1)} = x \cdot 1 = x.$$

Vi har alltså visat att $x^{ed} \equiv x \pmod{p}$ även när p inte delar x. Vi vill även visa att $x^{ed} \equiv x \pmod{q}$. Detta görs på samma sätt som när $x^{ed} \equiv x \pmod{p}$ visades. Den enda skillnaden är att vi byter plats på p och q.

Sedan tidigare vet vi att, om a, b och m är heltal, gäller det att om $a \equiv b \pmod{m}$ så har vi att differensen $a - b$ är delbar med m. Om vi

applicerar detta på vårt bevis har vi att eftersom $x^{ed} \equiv x \pmod{p}$ så är $x^{ed} - x$ delbart med p. Vi har då även att eftersom $x^{ed} \equiv x \pmod{q}$ så är $x^{ed} - x$ även delbart med q.

Slutligen har vi att eftersom $x^{ed} - x$ är delbart både med p och q samt att $n = p \cdot q$ så är $x^{ed} - x$ även delbart med n. Eftersom $x^{ed} - x$ är delbart med n gäller det att

$$x^{ed} \equiv x \pmod{n}$$

och det var detta vi från början skulle bevisa.

□

Kapitel 4

Primtalstester

4.1 Euklides sats

RSA-kryptografi bygger på att en snabb dator snabbt ska kunna hitta ett primtal, p eller q, innehållande omkring $100 - 200$ siffror. För att detta ska kunna ske behövs en snabb metod för att testa om ett cirka $100 - 200$ siffror stort tal är ett primtal. För det första måste vi veta att det finns så stora primtal. Detta vet vi är sant eftersom *Euklides sats* säger att antalet primtal är oändligt många. Nedan följer Euklides sats samt en förenklad förklaring av dess bevis.

Sats 4.1.1. *Det finns oändligt många primtal.*

Bevis. Detta är inget generellt bevis utan är bara ett exempel som visar hur grundtankarna i själva beviset ser ut. Vi ska alltså visa att det finns oändligt många primtal. Vi börjar därför med att anta motsatsen, det vill säga att det bara finns ett ändligt antal primtal. Vi antar att endast talen 2, 3, 5 och 7 är primtal. Om vi multiplicerer dessa med varandra och adderar 1 får vi ett tal vi kallar A.

$$A = (2 \cdot 3 \cdot 5 \cdot 7) + 1 = 210 + 1 = 211.$$

Vi vet från kapitel 1.5 att alla positiva heltal kan, på endast ett sätt, skrivas som en produkt av primtal. Vi kan därför skriva A som en produkt av primtal. Alltså kan A skrivas som ett visst primtal p multiplicerat med ett visst heltal B och då är $A = p \cdot B$. För att kunna forstsätta beviset måste vi även skriva $A - 1 = 210$ som en produkt med faktorn p. Vi ser i uppbyggnaden av talet A att $A - 1$ har primtalet p som faktor. Vi kan då skriva att $A - 1 = p \cdot C$ där C är ett heltal som inte är lika med B.

Vi utför några omskrivningar

$$A - 1 = p \cdot C \Leftrightarrow 1 = A - p \cdot C.$$

Sätt sedan in $A = p \cdot B$ och vi får

$$1 = p \cdot B - p \cdot C = p \cdot (B - C).$$

För att $1 = p \cdot (B - C)$ ska bli uppfyllt måste $p = 1$ och $(B - C) = 1$. Men detta kan inte stämma eftersom att p är ett primtal och primtal är större än 1. Vi har därför fått en motsägelse och det finns därmed fler primtal än 2, 3, 5 och 7 som vi antog i början.

Oavsett hur många primtal vi börjar beviset med kommer vi alltid komma fram till att det finns minst ett primtal till utöver de vi startade med. Detta innebär i slutändan att det finns oändligt många primtal och det var det vi skulle bevisa.

□

4.2 Fermat-testet

Ett annat problem som måste utredas är hur stora skillnader det är mellan stora primtal. Det kanske inte finns några primtal med 50-200 siffror. Då blir det stora problem att hitta primtal i rätt storlek och RSA-kryptografin blir inte lika användbar. Som tur är finns en sats som heter *primtalssatsen* som säger att man, statistiskt sett, kan förvänta sig att finna ett primtal i ett intervall med längden $\ln(n)$ runt ett stort heltal n. Vi har till exempel att runt det 100 siffror långa talet 10^{100} kan vi, statistiskt sett, förvänta oss att hitta ett primtal inom ett $\ln(10^{100}) \approx 230$ långt intervall runt 10^{100}.

Om vi har ett tal N som vi vill undersöka om det är ett primtal så finns det ett test som heter *Fermat-testet*. Talet N klarar testet om

$$b^{N-1} \equiv 1 \pmod{N}, \text{ där } 1 < b < N \text{ och } b \text{ är ett heltal.}$$

Vi har tidigare från Fermats lilla sats att $x^{p-1} \equiv 1 \pmod{p}$ där p är ett primtal. Likheterna vi ser mellan Fermats lilla sats och Fermat-testet gör att vi kan dra slutsatsen att alla primtal klarar sig igenom Fermat-testet. Problemet är att även vissa sammansatta tal klarar sig igenom Fermat-testet. Slutsatsen av Fermat-testet blir att N är något av följande:

- Definitivt inte primtal
- Kanske primtal

För att med större sannolikhet kunna säga att ett tal N är ett primtal kan ett *probabilistiskt primalitetstest* göras. Det går ut på att vi slumpmässigt väljer många olika heltalsvärden på b, dock alltid mellan 1 och N. Därefter utsätter vi flera gånger N för Fermat-testet, men vid varje test har vi olika värde på b. Om det vid något test inträffar att N inte klarar testet innebär det att N är ett sammansatt tal, alltså inget primtal. Ju fler test med olika b som N klarar desto större är sannolikheten att N är ett primtal. Sannolikheten att ett sammansatt tal N klarar Fermat-testet är inte större än 0,5. Detta betyder att sannolikheten är minst $1 - 0{,}5^n$ att ett sammansatt tal N inte klarar testerna, där n är antalet test. Vid exempelvis $n = 10$ blir sannolikheten $1 - 0{,}5^{10} \approx 0{,}999$. Om vi utsätter N för 100 test, och N klarar samtliga test, kommer sannolikheten vara extremt stor att N är ett primtal. Det finns dock ett annat primtalstest som med med total säkerhet avgör om talet är ett primtal. Detta primtalstest heter AKS-algoritmen och är generellt, villkorslöst och tidsåtgången är polynomiell med avseende på antalet siffror i talet som testas. För vidare läsning om detta hänvisar vi till *PRIMES is in P*.

4.3 RSA-kryptografins (o)säkerhet

En förutsättning för att RSA-kryptografi ska vara säkert är att det ska vara omöjligt att inom rimlig tid kunna faktorisera det omkring 200 siffror långa talet n till $p \cdot q$. Om p eller q är känt går det genom några beräkningar att få fram dekrypteringstalet d och RSA-kryptografin blir då oanvändbar. Än så länge finns det inga kända algoritmer som inom rimlig tid klarar att faktorisera n. Den lilla osäkerhet som finns kring RSA-kryptografin beror alltså på att ingen teoretiskt har bevisat att riktigt snabba faktoreringsalgoritmer inte finns för så stora sammansatta tal. Skulle dock en sådan algoritm dyka upp så kommer RSA-kryptografin bli obrukbar och matematiker får försöka klura ut en ny krypteringsalgoritm.

Kapitel 5

Facit

1. a) $26 = 2 \cdot 11 + 4$ b) $9 = 3 \cdot 3 + 0$ c) $16 = 5 \cdot 3 + 1$

2. a) 2 b) 15

3. a) $1, 2, 3, 4, 6, 8, 12, 24$ b) $1, 29$

4. 11 och 13

5. a) 1, relativt prima b) 31, ej relativt prima c) 18, ej relativt prima

6. a) Relativt prima b) Ej relativt prima c) Relativt prima

7. a) $x = -1 \;\; y = 3$ b) $x = -7 \;\; y = 5$ c) $x = -5 \;\; y = 14$
d) $x = 9 \;\; y = -58$

8. a) $d = 3$ b) $d = 23$

9. a) 1001_2 b) 10101_2 c) 1000100_2

10. a) Nej b) Ja

11. a) 0 b) 2 c) 29 d) 8 e) 0

12. a) 9 b) 11 c) 8 d) 130

13. a) $n = 91 \;\; m = 72 \;\; d = 31$ b) $n = 187 \;\; m = 160 \;\; d = 131$

14. a) 47 b) 150

15. a) 5 b) 7

16. a) $n = 77 \quad m = 60 \quad d = 53 \quad y = 41$

17. Om du får tillbaka ditt x efter dekryptering har du lyckats

18. Kontrollera med din kompis om du fick fram rätt meddelande

19. 16

20. En av oändligt många lösningar är $\{1,3,5,9,11,13\}$