TAMPEREEN YLIOPISTO
Pro gradu -tutkielma

Aki-Matti Luoto

Lukuteorian sovelluksia tiedon salauksessa

Matematiikan, tilastotieteen ja filosofian laitos
Matematiikka
Huhtikuu 2006

Tampereen yliopisto

Matematiikan, tilastotieteen ja filosofian laitos

LUOTO, AKI-MATTI: Lukuteorian sovelluksia tiedon salauksessa

Pro gradu -tutkielma, 27 s., 3 liites.

Matematiikka

Huhtikuu 2006

Tiivistelmä

Tutkielman alkuosassa käydään läpi lukuteorian perusteita ja loppuosassa lukuteoriaa sovelletaan kahden salausmenetelmän yhteydessä. Lukuteorian perusteet ovat pohjana hieman syvemmälle menevään materiaaliin, jotka tarvitaan, että voidaan ymmärtää RSA -salausmetodin salat. Tutkielmassa tarkastellaan myös muutamia alkulukutestejä.

Lähdekirjallisuutena on käytetty kahta kirjaa, jotka ovat Kenneth H. Rosenin Elementary number theory and "it's applications"ja tekijöiden Thomas H.Cormen, Charles E. Leiserson ja Ronald L. Rivest kirjoittama "Introduction to algorithms". Jälkimmäinen on näkökulmaltaan ohjelmoijille suunnattu kirja, kun taas edellinen on lukuteorian perusteita ja soveltamista käsittelevä teos.

Sisältö

1 Johdanto

Tiedon salaus on nykypäivää tieto- ja matkapuhelinverkoissa tapahtuvassa tiedonsiirrossa. Usein verkossa kulkevaa tietoa on tarve suojata ylimääräisiltä osapuolilta. Tällaista salattavaa tietoa ovat tyypillisesti esimerkiksi pankkitapahtumat. Matkapuhelinliikennekin on salattu menetelmällä, joka perustuu lukuteoriaan. Nykyään paljon käytetty RSA -salausmenetelmä kiinnosti ja näin ollen se oli luontava valinta Pro Gradu -tutkielmani aiheeksi.

Tutkielman alkuosassa käydään läpi lukuteorian perusteita ja loppuosassa lukuteoriaa sovelletaan kahden salausmenetelmän yhteydessä. Lukuteorian perusteet ovat pohjana hieman syvemmälle menevään materiaaliin, jotka tarvitaan, että voidaan ymmärtää RSA -salausmetodin salat. Perusajatuksena on, että matematiikan lukiotiedot omaava henkilö pystyy ymmärtämään RSA -menetelmän tämän tutkielman perusteella.

Tutkielman lähdekirjallisuutena on käytetty kahta kirjaa, jotka ovat Kenneth H. Rosenin "Elementary number theory and it's applications"ja tekijöiden Thomas H.Cormen, Charles E. Leiserson ja Ronald L. Rivest kirjoittama "Introduction to algorithms". Jälkimmäinen on näkökulmaltaan ohjelmoijille suunnattu kirja, kun taas edellinen on lukuteorian perusteita ja soveltamista käsittelevä opus.

2 Lukuteorian perusteita

Tähän luvun alkuun on koottu lukuteorian perusmääritelmiä ja lauseita esimerkkeineen. Ensin esitetään ja todistetaan jaollisuus, kongruenssi ja lukuteorian peruslauseita. Luvun loppupuolella esiteltävät lauseet liittyvät läheisesti RSA -salausmenetelmään.

2.1 Jaollisuus ja kongruenssi

Määritelmä 2.1 *[1, s. 36] Jos a ja b ovat kokonaislukuja ja $a \neq 0$, niin luku a jakaa b:n, jos on olemassa kokonaisluku c siten, että $b = ac$. Voidaan sanoa myös, että c on a:n tekijä.*

Jos a jakaa b:n, niin merkitään $a \mid b$. Jos a ei jaa lukua b, niin merkitään $a \nmid b$.

Jaollisuuden määritelmän perusteella huomoida, että mikä tahansa kokonaisluku jakaa nollan.

Lause 2.1 *[1, s. 37] Oletetaan, että a, b, c ja $d \in \mathbf{Z}$, ja $c \mid a$ ja $c \mid b$. Silloin $c \mid (ma + nb)$.*

Todistus. Koska $c \mid a$ ja $c \mid b$, niin on olemassa kokonaisluvut e ja f siten, että $a = ce$ ja $b = cf$. Siis $ma+nb = mce+ncf = c(me+nf)$. Siis $c \mid (ma+nb)$. □

Jatkossa tarvitsemme myös jakoalgoritmia, jonka todistus ohitetaan.

Lause 2.2 Jakoalgoritmi *[1, s. 37]. Jos a ja b ovat kokonaislukuja siten, että $b > 0$, niin on olemassa yksikäsitteiset luvut q ja r siten, että $a = bq + r$, jossa $0 \leq r < b$*

Esimerkki 2.1 *Olkoot $a = 25$ ja $b = 10$. Tällöin $q = 2$ ja $r = 5$, ja siis $25 = 2 \cdot 10 + 5$.*

Määritelmä 2.2 *[1, s. 120] Olkoon m positiivinen kokonaisluku, ja olkoot a ja b kokonaislukuja. Silloin a on kongruentti luvun b kanssa modulo m, jos $m \mid (a - b)$.*

Jos a on kongruentti b:n kanssa modulo m, niin kirjoitamme $a \equiv b \pmod{m}$. Jos taas a ei ole kongruentti b:n kanssa modulo m, niin kirjoitamme $a \not\equiv b$ ja tällöin $m \nmid (a - b)$.

Esimerkki 2.2 $14 \equiv 4 \pmod{5}$, *koska* $5 \mid 14 - 4$. *Samoin* $-27 \equiv 5 \pmod{8}$, *koska* $8 \mid -27 - 5$.

Lause 2.3 *[1, s. 120] Olkoot a ja b ovat positiivisia kokonaislukuja. Tällöin $a \equiv b \pmod{n}$, jos ja vain jos on olemassa $k \in \mathbf{Z}$ siten, että $a = b + km$.*

Todistus. Jos $a \equiv b \pmod{m}$, niin $m \mid (a - b)$. Tällöin on olemassa $k \in \mathbf{Z}$, siten että $km = a - b$, josta saadaan $a = b + km$

□

Seuraavaksi esitetään ja todistetaan kongruenssirelaation refleksiivisyys, symmetrisyys ja transitiivisuus [1, s. 120].

Lause 2.4 *Olkoot a, b ja c positiivisia kokonaislukuja. Tällöin on voimassa:*

(i) *Jos a on kokonaisluku, niin $a \equiv a \pmod{m}$.*

(ii) *Jos a ja b ovat kokonaislukuja siten, että $a \equiv b \pmod{m}$, niin $b \equiv a \pmod{m}$.*

(iii) *Jos a, b ja c ovat kokonaislukuja siten, että $a \equiv b \pmod{m}$ ja $b \equiv c \pmod{m}$, niin $a \equiv c \pmod{m}$.*

Todistus.

(i) Koska $m \mid (a - a)$, niin $a \equiv a \pmod{m}$.

(ii) Jos $a \equiv b \pmod{m}$, tällöin $m \mid (a - b)$. Siis on olemassa $k \in \mathbf{Z}$ siten, että $km = a - b$, eli $(-k)m = b - a$, eli $m \mid (b - a)$. Siis $b \equiv a \pmod{m}$.

(iii) Jos $a \equiv b \pmod{m}$ ja $b \equiv c \pmod{m}$. niin $m \mid (a-b)$ ja $m \mid (b-c)$. Siis on olemassa kokonaisluvut k ja l siten, että $km = a - b$ ja $lm = b - c$. Tällöin voidaan kirjoittaa, $a - c = (a - b) + (b - c) = (k + l)m$. Siis $m \mid (a - c)$ ja $a \equiv c \pmod{m}$.

□

Edellisen lauseen perusteella kongruenssi on ekvivelenssirelaatio, joten kokonaislukujen joukko voidaan jakaa *jäännösluokkiin modulo m*, jolloin jokaisen jäännösluokan luvut modulo m ovat keskenään kongruentteja modulo m.

Esimerkki 2.3 *Neljä jäännösluokkaa modulo 4 ovat*

$$\cdots \equiv -8 \equiv -4 \equiv 0 \equiv 4 \equiv 8 \equiv \cdots \pmod{4}$$
$$\cdots \equiv -7 \equiv -3 \equiv 1 \equiv 5 \equiv 9 \equiv \cdots \pmod{4}$$
$$\cdots \equiv -6 \equiv -2 \equiv 2 \equiv 6 \equiv 10 \equiv \cdots \pmod{4}$$
$$\cdots \equiv -5 \equiv -1 \equiv 3 \equiv 7 \equiv 11 \equiv \cdots \pmod{4}.$$

Lukuteoreettisissa esimerkeissä modulo-operaattorin sijasta saatetaan käyttää ”on yhtä kuin”-merkkiä. Tällöin kyseessä on pienin ei-negatiivinen luku, joka on kongruentti tietylle luvulle. Toisin sanoen kyse on jakojäännöksestä. Sama matemaattisesti esitettynä: olkoon m positiivinen kokonaisluku ja olkoot

a, b ja m kokokonaislukuja. Tällöin a voidaan esittää muiden lukujen avulla seuraavasti: $a = bm + r$, jossa $0 \leq r \leq m - 1$. r on nyt pienin ei-negatiivinen luku, joka on kongruentti r:lle, ja tämä merkitään $a \mod m = r$. Esimerkiksi $11 \mod 4 = 3$ ja $-17 \mod 6 = 1$.

Määritelmä 2.3 *[1, s. 120] Täydellinen jäännössysteemi modulo m on kokonaislukujen joukko siten, että joukon jokainen luku on kongruentti modulo m täsmälleen yhden tämän joukon alkion kanssa.*

Esimerkki 2.4 *Täydellisiä jäännössysteemejä modulo* 4 *ovat kokonaislukujen joukot* $0, 1, 2, 3$ *ja* $-4, 1, 2, 7$.

Seuraavaksi esitetään ja todistetaan kongruenssien yhteen-, vähennys- ja kertolasku [1, s. 122].

Lause 2.5 *Olkoot a, b, c ja m kokonaislukuja siten, että $m > 0$ ja olkoot voimassa $a \equiv b \pmod{m}$ ja $c \equiv d \pmod{m}$. Tällöin on voimassa:*

(i) $a + c \equiv b + d \pmod{m}$,

(ii) $a - c \equiv b - d \pmod{m}$,

(iii) $ac \equiv bd \pmod{m}$.

Todistus.

(i) Koska $a \equiv b \pmod{m}$, niin $m \mid a - b$, eli $m \mid (a + c) - (b + c)$. Siis $a + c \equiv b + d \pmod{m}$.

(ii) Koska $a \equiv b \pmod{m}$, niin $m \mid a - b$, eli $m \mid (a - c) - (b - c)$. Siis $a - c \equiv b - d \pmod{m}$.

(iii) Jos $a \equiv b \pmod{m}$ ja $b \equiv c \pmod{m}$, niin $m \mid (a - b)$ ja $m \mid (b - c)$. Siis on olemassa kokonaisluvut k ja l siten, että $km = a - b$ ja $lm = b - c$. Tällöin voidaan kirjoittaa, $a - c = (a - b) + (b - c) = (k + l)m$. Siis $m \mid (a - c)$ ja $a \equiv c \pmod{m}$.

□

Kongruenssiyhtälöiden jakaminen ei toimi ihan yhtä triviaalisti. Kongruenssilausekkeet voidaan jakaa kokonaisluvulla joissain tapauksissa, mutta se edellyttää tietyn ehdon täyttymistä. Ennen kuin tarkastellaan tätä ehtoa, määritellään suurin yhteinen tekijä.

Määritelmä 2.4 *[1, s. 74] Suurin yhteinen tekijä luvuille a ja b on suurin mahdollinen luku, joka jakaa molemmat luvut a ja b.*

Lukujen a ja b Suurinta yhteistä tekijää merkitään (a, b) tai syt(a, b). Määrittelemme myös $(0, 0) = 0$.

Esimerkki 2.5 $(8, 4) = 4, (-10, 5) = 5, (97, 50) = 1.$

Edellisen esimerkkien luvut on laskettavissa ilman tietokonetta, mutta suurien lukujen ollessa kyseessä suurin yhteinen tekijä voidaan laskea Eukleideen algoritmilla, joka esitetään ja todistetaan myöhemmin.

Lause 2.6 *[1, s. 122] Jos a, b, c ja m ovat positiivisia kokonaislukuja siten, että $m > 0$ ja $d = (c, m)$, ja $ac \equiv bc \pmod{m}$, niin $a \equiv b \pmod{m/d}$.*

Todistus. Jos $ac \equiv bc \pmod{m}$ niin $m \mid (ac-bc) = c(a-b)$. Siis on olemassa positiivinen kokonaisluku k siten, että $c(a - b) = km$. Jaetaan molemmat puolet d:llä, jolloin saadaan $(c/d)(a - b) = k(m/d)$. Koska $d = (c, m)$, niin myös $(m/d, c/d) = 1$. Eli $m/d \mid (a - b)$. Siis $a \equiv b \pmod{m/d}$.

□

Esimerkki 2.6 *Tarkastellaan kongruenssia* $150 = 5 \cdot 30 \equiv 9 \cdot 30 = 270 \pmod{20}$. *Kongruenssi voidaan jakaa luvulla 30, koska* $(10, 30) = 10$ *ja saadaan* $5 \equiv 9 \pmod{2}$. *Samoin kongruenssi* $100 \equiv 70 \pmod{15}$ *voidaan jakaa luvulla 10 ja saadaan* $10 \equiv 7 \pmod{3}$, *koska* $(10, 15) = 5$.

Seuraus 2.6.1. [1, s. 120] Jos a, b, c ja m ovat kokonaislukuja siten, että $m > 0$, $(c, m) = 1$ ja $ac \equiv bc \pmod{m}$, niin $a \equiv b \pmod{m}$.

2.2 Modulaarinen potenssiin korotus

Tässä luvussa esittelemme algoritmin, jonka avulla tietokone pystyy laskemaan suuristakin luvuista kongruensseja varsin vähällä laskennalla. Nämä suuret luvut sovelluksissa ovat usein potenssiesityksiä. Tarkastellaan asiaa esimerkin kautta laskemalla kongruenssin $3^{120} \pmod{267}$ suurin ei-negatiivinen ratkaisu käyttämällä apuna lausetta 2.5. eli kongruenssin kertolaskusääntöä.

Esimerkki 2.7

$$3 \equiv 3 \pmod{267}$$

$$3^2 \equiv 9 \pmod{267}$$

$$3^4 \equiv 81 \pmod{267}$$

$$3^8 \equiv 153 \pmod{267}$$

$$3^{16} \equiv 153 \cdot 153 \equiv 180 \pmod{267}$$

$$3^{32} \equiv 180 \cdot 180 \equiv 93 \pmod{267}$$

$$3^{64} \equiv 93 \cdot 93 \equiv 105 \pmod{267}$$

Siis

$$3^{120} = 3^{64} \cdot 3^{32} \cdot 3^{16} \cdot 3^8$$

$$\equiv 105 \cdot 93 \cdot 180 \cdot 153 \pmod{267} = 93.$$

2.3 Alkuluvut

Alkuluvut ovat luonnollisia jaottomia lukuja, joita on olemassa ääretön määrä [1, s. 65] . Viisi pienintä alkulukua ovat 2,3,5,7 ja 11. On olemassa erilaisia lukuteoreettisia tapoja, joilla voidaan testailla tietyissä rajoissa, onko luku alkuluku vai ei, mutta nykyään käytössä olevalla matemaattisella tiedolla ja tietokoneen prosessorivoimalla ei voida prosessoida täydellistä alkulukujonoa kovinkaan pitkälle. Periaatteessa pelkällä - tässä luvussa esitettävällä - Eukleideen algoritmillä voitaisiin testata kaikkien lukujen jaottomuus, mutta suurien lukujen ollessa kyseessä tämä ei ole mahdollista kohtuullisessa ajassa. Yhdistelemällä sopivasti joitain alkulukutestejä voidaan isoistakin luvuista päätellä hyvin suurella todennäköisyydellä, että ovatko ne alkulukuja. Modulaarista potenssiinkorotus algoritmia käyttäen voidaan kätevästi - alustavasti

- selvittää lukujen alkulukuominaisuutta [2, s. 839] ja tämä onnistuu vieläpä suhteellisen vähällä laskennalla. Kyseinen alkulukutesti ei ole kuitenkaan täysin aukoton, vaan se hyväksyy alkuluvuiksi lukuja, jotka todellisuudessa eivät ole alkulukuja. Tällaisia lukuja kutsutaan *Carmichaelin luvuiksi*, joita on kuitenkin suhteellisen harvassa. RSA- salauksen toteuttamiseen tarvitaan kaksi suurta alkulukua, jotka riittävän suurella todennäköisyydellä voidaankin todeta alkuluvuiksi Millerin-Rabinin alkulukutestillä [2, s. 841].

Määritelmä 2.5 *[1, s. 64] Alkuluku on luku, joka on suurempi kuin yksi ja on jaollinen vain itsellään ja luvulla yksi.*

Esimerkki 2.8 *Luvut 2, 3, 5, 53, 97 ovat alkulukuja, koska niitä ei voi jakaa tekijöihin. Luvut 4, 9, 100 eivät ole alkulukuja, koska* $4 = 2 \cdot 2$, $9 = 3 \cdot 3$ *ja* $50 \cdot 2 = 100$.

Määritelmä 2.6 *[1, s. 64] Jos luku ei ole alkuluku, se on yhdistetty luku.*

Määritelmä 2.7 *[1, s. 74] Lukuja a ja b sanotaan suhteellisiksi alkuluvuiksi, mikäli* $(a, b) = 1$.

Esimerkki 2.9 *Luvut* 50 *ja* 9 *ovat suhteellisia alkulukuja, koska* $(50, 9) = 1$.

2.4 Eukleideen algoritmi

Noin kolme sataa vuotta ennen ajanlaskun alkua kreikkalainen matemaatikko esitti julkaisussaan Elements algoritmin [2, s. 851], jolla mikä tahansa luku voidaan jakaa tekijöihin. Seuraavaksi esitetään ja todistetaan lause.

Lause 2.7 Eukleideen algoritmi. *[1, s. 80] Olkoot* $r_0 = a$ *ja* $r_1 = b$ *positiivisia kokonaislukuja siten, että* $a \geq b > 0$. *Jakoalgoritmia käyttäen saadaan* $r_j = r_{j+1}q_{j+1} + r_{j+2}$ *siten, että* $0 < r_{j+2} < r_{j+1}$, *kun* $j = 0, 1, 2, ...; n - 2$ *ja* $r_{n+1} = 0$, *silloin* $(a, b) = r_n$ *on viimeinen ei-negatiivinen jäännös.*

Ennen Eukleideen algoritmin todistusta todistamme tarvittavan lemman.

Lause 2.8 *[1, s. 81] Jos c ja d ovat kokonaislukuja siten, että* $c = dq + r$, *missä q ja r ovat kokonaislukuja, niin* $(c, d) = (d, r)$.

Todistus. Jos kokonaisluku e jakaa luvut c ja d, niin koska $r = c - dq$, niin lauseen 2.1. perusteella $e \mid r$. Jos $e \mid d$ ja $e \mid r$, niin koska $c = dq + r$, niin edelleen lauseseen 2.1. nojaten saadaan $e \mid c$. Koska nyt lukupareilla c, d ja d, r on samat yhteiset tekijät, täytyy olla $(c, d) = (d, r)$.

□

Nyt voidaan todistaa Eukleideen algoritmi.

Todistus [1, s. 81]. Olkoot $r_0 = a$ ja $r_1 = b$ positiivisia kokonaislukuja siten, että $a \geq b$. Jakoalgoritmia toteuttaen saamme

$$
\begin{aligned}
r_0 &= r_1 q_1 + r_2 0 \leq r_2 < r_1, \\
r_1 &= r_1 q_2 + r_2 0 \leq r_3 < r_2, \\
&\dots \\
r_{j-2} &= r_{j-1} q_{j-1} + r_j 0 \leq r_j < r_{j-1}, \\
&\dots \\
r_{n-4} &= r_{n-3} q_{n-3} + r_{n-2} 0 \leq r_j < r_{j-1}, \\
r_{n-3} &= r_{n-2} q_{n-2} + r_{n-1} 0 \leq r_j < r_{j-1}, \\
r_{n-2} &= r_{n-1} q_n + r_n 0 \leq r_j < r_{j-1}, \\
r_{n-1} &= r_n q_n.
\end{aligned}
$$

□

Esimerkki 2.10 *Lasketaan* $(532, 201)$ *ja* $(10, 5)$ *Eukleideen algoritmilla. Suurimmat yhteiset tekijät on lihavoituna.*

$$
\begin{aligned}
532 &= 2 \cdot 201 + 130 \\
201 &= 1 \cdot 130 + 71 \\
130 &= 1 \cdot 71 + 59 \\
71 &= 1 \cdot 59 + 12 \\
29 &= 2 \cdot 12 + 5 \\
12 &= 2 \cdot 5 + 2
\end{aligned}
$$

$$5 = 2 \cdot 2 + 1$$
$$2 = 2 \cdot \mathbf{1} + 0$$

$$10 = 2 \cdot \mathbf{5}$$

Eukleideen algoritmillä voidaan selvittää, onko jokin alkuluku vai yhdistetty luku. Oletetaan, että luku p on alkuluku. Tällöin

$$(p, p-1) = (p, p-2) = (p, p-3) = \cdots = (p, 3) = (p, 2) = 1.$$

Jos luku p ei ole alkuluku, niin jokin seuraavista suurimmista yhteisistä tekijöistä

$$(p, p-1), (p, p-2), (p, p-3), \ldots, (p, 3), (p, 2)$$

on suurempi kuin 1.

Mitä suurempi on luku, jonka alkulukuominaisuus halutaan tietää, sen työläämpi lasku on. Jos selvitetään luvun p alkulukuominaisuutta edellisen kappaleen tietojen perusteella, pitää Eukleideen algoritmi suorittaa $p-2$ kertaa ja mitä suurempia ovat luvut, joista suurin yhteinen tekijä lasketaan, sen työläämpi Eukleideen algoritmi on.

Suurinta yhteistä tekijää laskettaessa voidaan laskentatyötä vähentää käyttämällä rekursiivista funktiota [2, s. 809]. Näin laskentatehtävä helpottuu paljon, mutta se ei muuta tosiasiaa, että isojen lukujen jako tekijöihin on erittäin työlästä.

2.5 Lineaarinen Diofantoksen yhtälö

Kahden muuttujan lineaariyhtälöitä kutsutaan Diofantoksen yhtälöiksi. Seuraavaksi esitellään ja todistetaan lauseita, jotka ovat esitietoina, jotta päästään otsikon asiaan. Lisäksi määritellään lineaarikombinaatio.

Lause 2.9 *[1, s. 75] Olkoot a, b ja c kokonaislukuja siten, että $(a, b) = d$. Silloin $(a/d, b/d) = 1$.*

Todistus. Oletetaan, että e on positiivinen kokonaisluku siten, että $e \mid (a/d)$ ja $e \mid (b/d)$. Tällöin on olemassa kokonaisluvut k ja l siten, että $a/d = ke$ ja

$b/d = le$, joten $a = dek$ ja $b = del$. Tällöin $de \mid a$ ja $de \mid b$. Koska $(a, b) = d$, niin $de \leq d$, eli $e = 1$. Siis $(a/d, b/d) = 1$.

□

Määritelmä 2.8 *[1, s. 75] Jos a ja b ovat kokonaislukuja, niin a:n ja b:n lineaarikombinaatio on muotoa ma + nb, jossa m ja n ovat kokonaislukuja.*

Lause 2.10 *[1, s. 75] Jos a ja b ovat kokonaislukuja, joista ainakin toinen eroaa nollasta, ja d = (a, b), niin luku d on pienin luku, joka voidaan muodostaa lukujen a ja b lineaarikombinaationa.*

Todistus. Olkoot a ja b kokonaislukuja, $a \neq 0$. Olkoon d pienin luku, joka voidaan antaa a:n ja b:n lineaarikombinaationa. Nyt

$$d = ma + nb,$$

jossa a ja b ovat kokonaislukuja. Jakoalgoritmin perusteella saamme

$$a = dq + r, 0 \leq r < d.$$

Edelleen

$$r = a - dq = a - q(ma + nb) = (1 - qm)a - qnp.$$

Siis r on a:n ja b:n lineaarikombinaatio. Koska $0 \leq r < d = ma + mb$ ja d pienin mahdollinen a:n ja b:n lineaarikombinaatio, niin r:n pitää olla 0, eli $d \mid a$. Vaihtamalla todistuksessa b:n ja a:n paikkaa, voimme vastaavasti todistaa, että $d \mid b$.

Vielä on todistettava, että d on nimenomaan suurin yhteinen tekijä eli $d = (a, b)$. Valitaan mielivaltainen c siten, että $c \mid a$ ja $c \mid b$. Koska $d = ma+nb$, niin lauseen 2.1 perusteella $c \mid d$.

□

Nyt voidaan tarkastella Diofantoksen yhtälöä käytännön ongelman kautta: opettajalla on rahapussissa 47 kolikkoa, 20:n sentin ja 50:n sentin kolikoina 1900 senttiä. Montako kappaletta hänellä on kutakin kolikkoa? Tähän kysymykseen voidaan vastata ratkaisemalla yhtälö $20x + 50y = 1900$. Kyseeessä kahden muuttujan lineaariyhtälö, joita kutsutaan Diofantoksen yhtälöiksi.

Tässä esimerkissä yhtälö on ratkeava, mutta aina Diofantoksen yhtälöt eivät ole ratkaistavissa. Lukuteorian keinoin on helppo selvittää, onko yhtälö ratkaistavissa vai ei.

Lause 2.11 *[1, s. 91] Jos a, b ja c ovat positiivisia kokonaislukuja siten, että $(a, b) = 1$ ja $a \mid bc$, niin $a \mid c$.*

Todistus. Koska $(a, b) = 1$, niin on olemassa kokonaisluvut x ja y siten, että $ax + by = 1$. Kertomalla yhtälö luvulla c, saadaan $acx + bcy = c$. Lauseen 2.1 nojalla $a \mid acx + bcy$, koska $acx + bcy$ on a:n ja bc:n lineaarikombinaatio ja toisaalta molemmat ovat jaollisia a:lla. Siis $a \mid c$.

□

Edellä esitetty Diafantoksen yhtälö $20x + 50y = 1900$ on siis ratkaistavissa lauseen perusteella, koska $(50, 20) = 10$ ja $10 \mid 1900$. Jos Diafantoksen yhtälöllä on ainakin yksi ratkaisu, muut ratkaisut saadaan seuraavaksi esitettävän lauseen avulla.

Lause 2.12 *[1, s. 113] Oletetaan, että a ja b ovat positiivisia kokonaislukuja siten, että $d = (a, b)$. Kahden muuttujan yhtälöllä $ax + by = c$ ei ole ratkaisua, jos $d \nmid c$. Jos $d \mid c$, niin yhtälöllä on ääretön määrä ratkaisuja. Yleisemmin: jos $x = x_0$, $y = y_0$ on yhtälön yksi ratkaisu, niin kaikki ratkaisut voidaan antaa muodossa*

$$x = x_0 + (b/d)n, y = y_0 - (a/d)n,$$

jossa $n \in \mathbf{Z}$.

Todistus. Olkoot x ja y sellaiset kokonaisluvut, että $ax + by = c$. Koska $d \mid a$ ja $d \mid b$, niin lauseen 2.1. nojalla myös $d \mid c$. Jos $d \nmid c$, niin yhtälöllä ei olisi kokonaislukuratkaisua.

Oletetaan, että $d \mid c$. Tällöin lauseen 2.8 mukaan on olemassa luvut s ja t siten, että

$$d = as + bt.$$

Koska $d \mid c$, on niin olen olemassa $e \in \mathbf{Z}$ siten, että $de = c$. Kertomalla yhtälö luvulla e, saadaan

$$c = de = (as + bt)e = a(se) + b(te).$$

Siis yksi ratkaisu yhtälölle on $x = x_0 = se$ ja $y = y_0 = te$.

Osoittaaksemme, että yhtälöllä on ääretön määrä ratkaisuja, merkitään $x = x_0 + (b/d)n$ ja $y = y_0 - (a/d)n$, jossa $n \in \mathbf{Z}$. Pari (x, y) on ratkaisu, koska

$$ax + by = ax_0 + a(b/d)n + by_0 - b(a/d)n = ax_0 + by_0 = c.$$

Näytämme vielä, että jokainen yhtälön $ax + bx = c$ ratkaiseva lukupari (x, y) on lauseessa esitettyä muotoa. Oletetaan, että yhtälön yksi ratkaisu on lukupari (x_0, y_0). Siis

$$ax_0 + by_0 = c,$$

eli

$$ax_0 + by_0 = ax + by,$$

josta edelleen saadaan

$$(ax + by) - (ax_0 + by_0) = 0.$$

Ottamalla a ja b yhteiseksi tekijöiksi saadaan

$$a(x - x_0) + b(y - y_0) = 0,$$

jolloin

$$a(x - x_0) = b(y_0 - y).$$

Jakamalla yhtälö vielä d:llä saadaan

$$(a/d)(x - x_0) = (b/d)(y_0 - y).$$

Lauseen 2.6. nojalla tiedämme, että $(a/d, b/d) = 1$. Tästä lauseen 2.7 nojalla $(a/d) \mid (y_0 - y)$. Eli on olemassa kokonaisluku n siten, että $(a/d)n = y_0 - y$, eli $y = y_0 - (a/d)n$. Sijoittamalla tämä y yhtälöön $a(x - x_0) = b(y_0 - y)$ saamme $a(x - x_0) = b(a/d)n$, josta seuraa

$$x = x_0 + (b/d)n.$$

□

Esitetyn ongelman eräs ratkaisu on $x_0 = 0$ ja $y_0 = 38$, jolloin lauseen 2.12 perusteella muut ratkaisut saadaan lausekkeista $x = 0 + n \cdot 5$ ja $y = 38 - 2 \cdot n$. Koska tähtävässä on voimassa myös ehto $x + y = 47$, niin yhtälöllä on ainoastaan yksi ratkaisu, joka on $x = 15, y = 32$.

2.6 Lineaarikongruenssi ja käänteisluku modulo m

Kongruenssia, joka ovat muotoa

$$ax \equiv b \pmod{m},$$

kutsutaan yhden muuttujan lineaariseksi kongruenssiyhtälöksi.

Lause 2.13 *[1, s. 131] Olkoon a, b, ja m kokonaislukuja siten, että $m > 0$ ja $(a, m) = d$. Jos $d \nmid b$, niin yhtälöllä $ax \equiv b \pmod{m}$ ei ole ratkaisuja. Jos $d \mid a$, niin yhtälöllä $ax \equiv b \pmod{m}$ on tarkelleen d kappaletta ei-kongruentteja ratkaisuja $\pmod{m}$.*

Todistus. Kongruenssiyhtälö $ax \equiv b \pmod{m}$ saadaan lauseen 2.1 nojalla muotoon $ax - my = b$. Lauseen 2.12 nojalla tällä Diofantoksen yhtälöllä ei ole ratkaisua, jos $d \nmid b$.

Jos $d \mid a$, niin ratkaisut ovat lauseen 2.12 perusteella muotoa

$$x = x_0 + (b/d)t, y = y_0 - (a/d)t.$$

Nähdään, että nyt kongruenssityhtälöllä $ax \equiv b \pmod{m}$ on ääretön määrä ratkaisuja, jotka ovat muotoa $x = x_0 + (b/d)t$. On selvitettävä siis, moniko ratkaisuista on ei-kongruentteja. Oletetaan, että $x_1 = x_0 + (m/d)t_1$ ja $x_2 = x_0 + (m/d)t_2$ ovat kongruentteja osaratkaisuja modulo m, jolloin saadaan

$$x_0 + (m/d)t_1 \equiv x_0 + (m/d)t_2 \pmod{m},$$

ja vähentämällä edelleen x_0 saadaan

$$(m/d)t_1 \equiv (m/d)t_2 \pmod{m}.$$

Koska $(m/d) \mid m$, niin $(m, m/d) = m/d$, eli kongruenssi voidaan jakaa lauseen 2.6. nojalla luvulla (m/d) ja tällöin saadaan

$$t_1 \equiv t_2 \pmod{d}.$$

Tästä nähdään, että kaikki ei-kongruentit ratkaisut saadaan lausekkeesta $x = x_0 + (m/d)t$, kun t käy läpi täydellisen jäännössyteemin modulo d.

□

Määritelmä 2.9 *[1, s. 132] Jos* $(a, m) = 1$*, niin kongruenssiyhtälön* $ax \equiv 1$ *ratkaisua* x *sanotaan* a*:n käänteisluvuksi modulo* m*.*

Esimerkki 2.11 *Koska kongruenssiyhtälön* $5x \equiv 1 \pmod{7}$ *ratkaisu on* $x = 3$*, niin luvun* 5 *käänteisluku modulo* 7 *on* 3*. Vastaavasti luvun* 3 *käänteisluku modulo* 7 *on* 5*.*

Lause 2.14 *[1, s. 133] Olkoon* p *alkuluku. Tällöin positiivinen luku* a *on itsensä käänteisluku jos, ja vain jos,* $a \equiv 1 \pmod{p}$ *tai* $a \equiv -1 \pmod{p}$*.*

Todistus. Jos $a \equiv 1 \pmod{p}$ tai $a \equiv -1 \pmod{p}$, niin $a^2 = 1 \pmod{p}$, eli a on itsensä käänteisluku modulo p.

Jos a on itsensä käänteisluku modulo p, niin $a^2 = a \cdot a \equiv \pmod{p}$. Koska $p \mid a^2 - 1 = (a-1)(a+1)$, niin joko $p \mid (a-1)$ tai $p \mid (a+1)$. Siis joko $a \equiv 1 \pmod{p}$ tai $a \equiv -1 \pmod{p}$.

□

2.7 Eulerin phi-funktio

Seuraavaksi esiteltävällä Euler Phi-funktio funktiolla on osansa RSA -salauksen toteutuksessa.

Määritelmä 2.10 *[1, s. 207] Euler Phi-funktiota merkitään* $\phi(n)$*, ja se kertoo, kuinka moni luvuista* $1, 2, ..., n$ *on suhteellinen alkuluku luvulle* n*.*

Esimerkki 2.12 $\phi(1) = 1$*, koska* $(1, 1) = 1$
$\phi(2) = 1$*, koska*$(1, 2) = 1$ *ja* $(2, 2) = 2$*,*
$\phi(3) = 2$*, koska* $(1, 3) = 1$,$(2, 3) = 1$,$(3, 3) = 3$*,*
$\phi(4) = 2$*, koska* $(1, 4) = 1$*,* $(2, 4) = 3$*,* $(3, 4) = 1$*,* $(4, 4) = 4$

Seuraavassa esitettyna taulukossa muutamia $\phi(n)$-funktion arvoja.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\phi(n)$	1	1	2	2	4	2	6	4	6	4	10	4	12	6

Taulukosta havaitaan, että kun n on alkuluku, niin $\phi(n) = n - 1$, mikä on seurausta alkuluvun määritelmästä. Alkuluvulla n on ainoastaan tekijät 1 ja n, ja siis ainoastaan $(n, n) = n$.

Määritelmä 2.11 *[1, s. 207] Aritmeettista funktiota f kutsutaan multiplikatiiviseksi funktioksi, jos $f(mn) = f(m)f(n)$ aina, kun m ja n ovat suhteellisia alkulukuja.*

Euler-Phi funktio on myös multiplikatiivinen funktio [1, s. 207], Seuraavassa asia esitettynä lauseessa, jonka todistus sivuutetaan.

Lause 2.15 *[1, s. 209] Olkoot positiiviset kokonaisluvut m ja n suhteellisia alkulukuja. Silloin $\phi(mn) = \phi(m)\phi(n)$.*

Esimerkki 2.13

$$\phi(12) = \phi(3)\phi(4) = 2 \cdot 2 = 4$$

Määritelmä 2.12 *[1, s. 202] Supistettu jäännössysteemi* $\pmod n$ *on $\phi(n)$ luvun joukko siten, että jokainen joukon luku on suhteellinen alkuluku luvun n kanssa, ja mikään joukon luvuista ei ole toisilleen kongruentteja* $\pmod n$.

Esimerkki 2.14 *Kokonaislukujen joukko $1, 5, 7, 11$ on supistettu jäännössysteemi* $\pmod{12}$, *koska $(1, 12) = (5, 12) = (7, 12) = (11, 12) = 1$, mitkään joukon luvut ole kongruentteja keskenään* $\pmod{12}$ *ja joukossa on lukuja $\phi(12)$ kappaletta alkioita.*

Jos jonkun supistetun jäännössysteemin $\pmod m$ luvut kerrotaan jollain kokonaisluvulla, joka on suhteellinen alkuluku luvun m kanssa, on saatu joukko myös supistettu jäännössysteemi $\pmod m$. Sama esitettynä lauseena, jonka todistus sivuutetaan. Lausetta tarvitaan Eulerin teoreeman todistamisessa.

Lause 2.16 *[1, s. 202] Jos kokonaislukujen joukko $r_1, r_2, \ldots, r_{\phi(n)}$ on supistettu jäännössysteemi* $\pmod n$ *ja $(a, n) = 1$, niin myös joukko $ar_1, ar_2, \ldots, ar_{\phi(n)}$ on supistettu jäännössysteemi* $\pmod n$.

Seuraavaksi esitellään ja todistetaan alkulukutesti, jossa phi-funktio esiintyy.

Lause 2.17 *[1, s. 208] p on alkuluku vain, ja vain jos $\phi(p) = p - 1$.*

Todistus. Jos p on alkuluku, niin jokainen luku, joka on pienempi kuin p, on suhteellinen alkuluku p:lle. Siis on olemassa $p - 1$ suhteellista alkulukua luvulle p, eli $\phi(p) = p - 1$.

Jos $\phi(p) = p-1$, ja p on yhdistetty luku, eli on olemassa luku d, joka jakaa luvun p, ja $1 < d < p$, eivätkä p ja d ole suhteellisia alkulukuja. Tarkastellaan lukuja 1,2,...,p-1. Näistä luvuista yksi, eli d, ei ole suhteellinen alku p:lle, ja näin ollen $\phi(p) \leq p - 2$. Tästä seuraa ristiriita olettamuksen kanssa, ja näin ollen p on alkuluku.

□

Seuraavaksi esitellään lause, jolla voidaan löytää Eulerin phi-funktion arvoja alkulukujen potensseille.

Lause 2.18 *[1, s. 208] Olkoon p alkuluku, ja $a \in \mathbf{Z}+$. Tällöin $\phi(p^a) = p^a - p^{a-1}$.*

Todistus. Luvuista $1, 2, ..., p^a$ ainoastaan luvulla p jaolliset luvut eivät ole suhteellisia alkulukuja p^a:lle. Merkitään näitä lukuja kp:llä, jossa $1 \leq k \leq p^{a-1}$. Siis on olemassa tarkalleen p^{a-1} kappaletta positiivisia kokonaislukuja, jotka eivät ole suhteellisia alkulukuja p:lle. Siten on $p^a - p^{a-1}$ positiivista kokonaislukua jotka ovat suhteellisia alkulukuja p:lle. Siis $\phi(p^a) = p^a - p^{a-1}$.

□

Esimerkki 2.15 $\phi(2^7) = 2^7 - 2^6 = 128 - 64 = 64$ *ja* $\phi(3^3) = 3^3 - 3^2 = 27 - 9 = 18$.

Lause 2.19 Eulerin teoreema. *[1, s. 203] Jos m on positiivinen kokonaisluku ja a on kokonaisluku siten, että $(a, m) = 1$, niin $a^{\phi(m)} \equiv 1 \pmod{m}$.*

Todistus. Olkoon kokonaislukujen joukko $r_1, r_2, \ldots, r_{\phi(m)}$ supistettu jäännössysteemi modulo n siten, että joukon luvut ovat pienempiä kuin m. Koska $(a, m) = 1$, niin lauseen 2.14. mukaan myös kokonaislukujen joukko $ar_1, ar_2, \ldots, ar_{\phi(n)}$ on supistettu jäännössysteemi modulo m. Tällöin positiivinen supistettu jäännössysteemi joukosta $ar_1, ar_2, \ldots, ar_{\phi(n)}$ sisältää luvut $r_1, r_2, \ldots, r_{\phi(m)}$ jossain järjestyksessä. Siis saadaan

$$ar_1, ar_2 \ldots ar_{\phi(m)} \equiv r_1 r_2 \ldots r_{\phi(m)} \pmod{m},$$

eli

$$a^{\phi(m)} r_1, r_2 \ldots r_{\phi(m)} \equiv r_1 r_2 \ldots r_{\phi(m)} \pmod{m}.$$

□

2.8 Wilsonin teoreema ja Fermat'n pieni lause

Seuraavaksi esittelemme kaksi mielenkiintoista kongruenssia [1, s. 185], jotka ovat tärkeitä lukuteorian sovelluksissa. RSA -salaus menetelmässä sovelletaan Fermat'm pientä lausetta. Wilsonin teoreemalla voidaan tietyissa rajoissa testata, onko jokin luku alkuluku vai ei.

Lause 2.20 *[1, s. 185] Jos p on alkuluku, niin* $(p-1)! \equiv -1 \pmod{p}$.

Ennen lauseen todistus tarkastelemme esimerkkiä, josta selviää todistuksen idea.

Esimerkki 2.16 *Olkoon luku $p = 11$. Koska*

$$4{\cdot}3 \equiv 1 \pmod{11}, 2{\cdot}6 \equiv 1 \pmod{11}, 7{\cdot}8 \equiv 1 \pmod{11}, 5{\cdot}9 \equiv 1 \pmod{11},$$

niin voidaan kirjoittaa:

$$(11-1)! = 1{\cdot}2{\cdot}3{\cdot}4{\cdot}5{\cdot}6{\cdot}7{\cdot}8{\cdot}9{\cdot}10 = 1{\cdot}(4{\cdot}3){\cdot}(2{\cdot}6){\cdot}(7{\cdot}8){\cdot}(5{\cdot}9){\cdot}10 = 1{\cdot}10 \equiv -1 \pmod{11}.$$

Todistus. Olkoon $p = 2$, jolloin $(p-1)! \equiv 1 \equiv -1 \pmod{2}$. Siis lause pitää paikkansa luvun p arvolla 2. Olkoon alkuluku p nyt suurempi kuin 2. Lauseen 2.13. nojalla jokaiselle kokonaisluvulle a välillä $[1, p-1]$ löytyy käänteisluku $\overline{a}$

(mod p) siten, että $0 \leq \overline{a} \leq p-1$ ja $a\overline{a} \equiv 1 \pmod{p}$. Lauseen 2.14 perusteella lukua p pienemmistä positiivisista kokonaisluvuista ainoastaan luvut 1 ja $p-1$ ovat itsensä vastalukuja modulo p. Nyt kaikki luvusta 2 lukuun $p-2$ voidaan järjestää pareiksi siten, että jokaisen parin tulo on kongruentti luvulle 1 modulo p, ja näin olleen saamme

$$2 \cdot 3 \ldots (p-3) \cdot (p-2) \equiv 1 \pmod{p}.$$

Kertomalla kongruenssi luvuilla 1 ja $(p-1)$ saadaan

$$(p-1)! = 1 \cdot 2 \cdot 3 \ldots (p-3) \cdot (p-2) \cdot (p-1) \equiv 1 \cdot (p-1) \equiv -1 \pmod{p}.$$

□

Wilsonin teoreema toimii myös toiseen suuntaan, eli jos $(p-1)! \equiv -1 \pmod{p}$, niin p on alkuluku. Lauseen todistus sivuutetaan, mutta mainittakoon, että aikakompleksisuutensa vuoksi kyseinen alkulukutesti ei ole kovinkaan käytännöllinen suurien lukujen testauksessa.

Lause 2.21 Fermat'n pieni lause. *[1, s. 187] Jos p on alkuluku, ja a on positiivinen kokonaisluku siten, että $p \nmid a$, niin $a^{p-1} \equiv 1 \pmod{p}$).*

Todistus. Olkoon a positiivinen kokonaisluku ja olkoon p jokin alkuluku, jolloin $(a, p) = 1$. Lauseen 2.19 mukaan $a^{\phi(m)} = 1$, eli $a^{\phi(p)} = 1$, josta saadaan edelleen $a^{p-1} \equiv 1 \pmod{p}$).

□

Fermat'n lauseesta on helppo johtaa lause, jonka avulla voidaan tutkia minkä tahansa positiivisen kokokonaisluvun alkulukuominaisuutta.

Lause 2.22 *[1, s. 188] Jos p on alkuluku ja a on positiivinen kokonaisluku, niin $a^p \equiv a \pmod{p}$.*

Todistus. Jos $p \nmid a$, niin Fermat'n lauseen perusteella $a^{p-1} \equiv 1 \pmod{p}$. Kun kerrotaan kongruenssi luvulla a, niin saamme $a^p \equiv a \pmod{p}$. Jos $p \mid a$, niin myös $p \mid a^p$, eli tällöin $a^p \equiv a \equiv 0 \pmod{p}$. Siis $a^p \equiv a \pmod{p}$ jos $p \nmid a$ tai $p \mid a$.

□

Muinaiset kiinalaiset olettivat edellä esitetyn lauseen toimivan myös käänteisesti [1, s. 192], mutta myöhemmin on todettu, että näin ei ole. Kiinalaiset matemaatikot olettivat, että jos on voimassa $2^n \equiv 2 \pmod{n}$, niin n on alkuluku. Suhteellisen usein tämä oletus toimiikin, mutta seuraavasta vastaesimerkistä nähdään, että aina näin ei ole.

Esimerkki 2.17 *Olkoon* $n = 645 = 3{\cdot}5{\cdot}43$*, eli* n *on yhdistetty luku. Kuitenkin on helppo todeta, että* $2^{644} \equiv 1 \pmod{645}$ *ja kun tämä kongruenssi kerrotaan kahdella, saadaan* $2^{645} \equiv 2 \pmod{645}$.

Edellisen esimerkin luku 645 on Carmichaelin luku ja samalla myös *pseudoalkuluku*. Pseudoalkulukun määritelmä esitetään seuraavaksi.

Määritelmä 2.13 *[1, s. 193] Olkoon luku* b *positiivinen kokonaisluku. Jos* n *on positiivinen kokonaisluku ja* $b^n \equiv b \pmod{n}$*, niin lukua* n *kutsutaan pseudoalkuluvuksi luvun* b *suhteen.*

3 Caesarin salaus

Tässä kappaleessa [1, s. 234] esitellään kryptografian käsitteitä ja periaatteita. Lisäksi seuraa palanen kryptografian historiaa; nimittäin seuraavaksi esitellään Caeserin salaus, joka oli käytössä Rooman imperiumilla. Matemaattisesti tämä salaus ei ole kovinkaan monimutkainen, mutta yhtäläisyyksiä löytyy uudempiin ja tehokkaampiin salausmenetelmiin. Nimittäin Caesarinkin salauksessa kirjaimet muutetaan numeroiksi, mikä onkin luontevaa, koska matematiikan avulla pyöritetään nimenomaan numeroita. Tarkasti ottaen luvun esimerkki ei ole Caesarin salaus, koska esimerkissä käytetään suomenkielen merkistöä, mutta periaate on täysin sama. Seuraavassa taulukossa on esitettynä suomenkielen aakkoset ja niitä vastaavat luvut.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

q	r	s	t	u	v	w	x	y	z	å	ä	ö
16	17	18	19	20	21	22	23	24	25	26	27	28

Merkkien määrä tietysti vaihtelee kielen mukaan, mutta luonteva esimerkkikieli on tässä tapauksessa suomi. Toki mukaan voisi ottaa välimerkkejä, tyhjän tai mitä tahansa muitakin merkkejä, mutta esimerkissä on käytössä suomalainen aakkosto, josta kirjaimia löytyy 29 kappaletta. Koska numerot - tai kirjaimet - voidaan järjestää 29! eri tavalla, niin voidaan todeta, että vastaavuudet voidaan toteuttaa 29! eri tavalla. Tämä voidaan yleistää, eli jos merkkejä on käytössä n kappaletta, niin erilaisia muunnoksia on $n!$ kappaletta. Tässä kappaleessa kuvatun tyyppisissä muunnoksissa merkkiä vastaava numero on aina sama, mikä merkitsee salauksen helppoa murrettavuutta.

Caesarin koodauksessa jokaista selväkielen merkkiä vastaa salatussa sanomassa merkki, joka on aakkosissa kolme pykälää edellä. Aakkosten loppupään kolme viimeistä kirjainta "kuvautuvat"aakkosten alkupäähän, kun selväkieli muutetaan salatuksi. Matemaattinen kuvaus kyseiselle muunnokselle voidaan antaa muodossa

$$C \equiv P + 3 \pmod{29}, 0 \leq C \leq 28,$$

jossa P on selväkielisen viestin merkkiä vastaava numeroarvo ja C on vastaava salatun tekstin kirjainta vastaava numero.

Caesarin koodauksen erikoispiirteenä on myös selväkieliviestin sanojen ryhmittely viiden merkin sanoiksi. Viimeinen sana voi olla ryhmittelyn jälkeen myös yhdestä neljään merkkiä pitkä, ja näin käykin, jos merkkien kokonaismäärä ei ole jaollinen viidellä.

Esimerkkinä muunnetaan viesti

SERGEI PITÄÄ KAALIKEITOSTA

Ryhmittelyn jälkeen viesti on

SERGE IPITÄ ÄKAALI KEITO STA

Vastaavat merkkien numeroarvot ovat

17 4 16 6 4 8 14 8 18 26 26 10 0 0 11 8 10 4 8 18 13 17 18 0

ja Caesarin muunnoksen jälkeen tilanne on

20 7 19 9 7 11 17 11 21 29 29 13 3 3 14 11 13 7 11 21 16 20 21 3

Muutetaan jälleen numerot niitä vastaaviksi merkeiksi

VHUJH LSLWÖ ÖODDP LOHLW RVWD

Salatun viestin purku onnistuu analogisesti, Caesarin salauksen purku onnistuu siirtymällä aakkosissa kolme askelta taaksepäin, kuitenkin niin , että C kuvautuu Ö:ksi jne. Ekvivalenssirelaation avulla esitettynä purku onnistuu seuraavasti [1, s. 237]

$$P \equiv C - 3 \pmod{29}, 0 \leq P \leq 28.$$

Caesarin salaus löytyy liitteestä yksi, toteutettuna C++ -kielisenä.

Caesarin salauksesta voidaan helposti tehdä muunnoksia vaihtamalla lisättävän luvun x arvoa kongruenssissa

$$P \equiv C - x \pmod{29}, 0 \leq P \leq 28, x \in \mathbf{Z} + .$$

Erilaisia salaustapoja edellisessä esimerkissä on 28, eli käytettävien merkkien määrä vähennettynä yhdellä, koska nollan käyttö ei tietenkään salaa selkokieltä. Koska mahdollisuuksia on näin vähän, vaihtoehdot on helppo kokeiltavissa. Lisättävän luvun arvoa vaihtamalla voidaan selvittää mikä vaihtoehto antaa selkokielistä tekstiä.

Salaus voidaan toki toteuttaa muullakin, kuin esimerkin mukaisella lukuteoriaa hyväksi käyttävällä tavalla. Vastaavuudet selkokieli ja salatun koodin välillä voidaan valita vaikkapa täysin mielivaltaisesti. Jos tällainen salaus yritetään purkaa oletuksella, että merkkejä on 29 kappaletta, niin saadaan $29! - 1$ eri vaihtoehtoa taulukolle, jossa on merkkien ja numeroiden vastaavuudet.

Tarkastellaan minkälainen urakka tietokoneelle olisi avata salattua tekstiä käymällä mahdollisia vaihtoehtoja läpi. Tietokoneella voi kokeilla kutakin vaihtoehtoa ja analysoida tekstin selkokielisyyttä. Järkevät tulokset ovat käytännössä tekstiä, jossa on selkokielisiä sanoja. Jos oletetaan, että tietokone käy läpi 3 miljoonaa (3000 MHz) vaihtoehtoa sekunnissa, niin karkeasti arvioituna koneelta kestää $29!/(3000 \cdot 10^6)$ sekuntia tehtävän suorittamiseen. Tämä aika on vuosiksi muutettuna suurusluokkaa $9,4 \cdot 10^{13}$, eli voidaan todeta sen olevan kohtuuton.

Ainoastaan raakaa prosessorivoimaa käyttäen tämän tyyppinen muunnos voi olla jo ylivoimainen tietokoneen ratkaistavaksi. Ongelmaa pystytään kuitenkin helpottamaan paljon, kun tiedetään salatun tekstin kieli. Koska muunnoksissa salatun merkin ja selkokielimerkin vastaavuus pysyvät samana, niin voidaan tarkastella merkkien esiintymistiheyksiä [1, s. 239]. Jos salatun tekstin

tiedetään olevan suomenkielistä, niin salatussa viestissä usein esiintyvä kirjain ei varmaankaan vastaa Z kirjainta. Toisaalta usein esiintyvä kirjan salatussa viestissä voisi olla A. Tälläisillä hyvillä arvauksilla voidaan helpottaa laskurakkaa huomattavasti ja näin ollen ongelma voidaan ratkaista tietokoneella tai jopa ilman tietokoneeen laskentaa. Ratkaisua helpottavia reunaehtoja voisi rakentaa myös sen perusteella, miten kirjaimet esiintyvät toistensa suhteen.

4 RSA

Tässä luvussa käsitellään RSA -salausta, joka kehitettiin 1970 luvulla [1, s. 260]. Nimensä tämä salausmenetelmä on saanut tekijöiltään: Rivest, Shamir ja Adleman.

4.1 RSA teoria

RSA -salaus perustuu salausavain pariin (e, n), jossa e on eksponentti, ja modulus n on suuri luku. Tämä luku n on kahden suuren alkuluvun tulo siten, että $n = pq$, jossa $(e, \phi(n)) = 1$. Jos merkitään d:llä luvun e käänteislukua $\pmod{n}$, niin luku $(p-1)\cdot(q-1)$ on jaollinen luvulla $de-1$. RSA -salauksen tehokkuus perustuu siihen, että tämä suuri luku n on erittäin työläs jakaa tekijöihin.

Käytettäessä RSA -salausmenetelmää, on tekstin merkit muunnettava niitä vastaaviksi numeroiksi samaan tapaa kuin Caesarin muunnoksessakin. Numeroista muodostetaan ryhmiä, joita merkitään P:llä. Ryhmän merkkien lukumäärä on parillinen, ja se riippuvainen käytetystä n:stä. Nyt kukin ryhmä P salataan, muodostamalla siitä ryhmä C.

$$E(P) = C \equiv P^e \pmod{n}, 0 < C < n$$

Salauksen purku edellyttää, että luvun e käänteisluku d modulo $\phi(n)$ tunnetaan. Tämä luku d on olemassa, koska $(e, \phi(n)) = 1$. Salatun numeroryhmän purku selkokielen numeroekvivalenteiksi tehdään seuraavasti:

$$D(C) \equiv C^d = (P^e)^d = P^{ed} = P^{k\phi(n)+1} \equiv (P^{\phi(n)})^k P \equiv P \pmod{n}.$$

missä $ed = k\phi(n) + 1$ jollain kokonaisluvulla k, jossa $ed \equiv 1 \pmod{\phi(n)}$ ja Eulerin teoreemaa käyttäen saamme $P^{\phi(n)} \equiv 1 \pmod{n}$, kun $(P, n) = 1$. Paria (d, n) sanotaan purkuavaimeksi.

4.2 RSA esimerkki

Seuraavaksi tarkastellaan RSA -menetelmän toimintaa esimerkin kautta. Valitaan ensin sopivat alkuluvut tekijöiksi luvulle n. Tässä käytetyt luvut eivät ole pienuutensa takia turvallisia, mutta esimerkkinä ne toimivat. Olkoot $p = 79$ ja $q = 53$, jolloin $n = pq = 4187$. Olkoon $e = 17$, jolloin $(e, \phi(n)) = 1$, eli luvun e käänteisluku d modulo $\phi(4187)$ on olemassa. Luku d saadaan ratkaisemalla kongruenssiyhtälö $17d \equiv 1 \pmod{\phi(4187)}$, mikä saadaan muotoon $17d \equiv 1 \pmod{4056}$, koska phi-funktio on multiplikatiivnen ja $78 \cdot 52 = 4056$. Kongruenssiyhtälö voidaan ratkaista ratkaisemalla Diofantoksen yhtälö $17x - 4056y = 1$. Ratkaisua saadaan Eukleideen algoritmiä käyttämällä.

$$4056 = 238 \cdot +10$$
$$17 = 1 \cdot 10 + 7$$
$$10 = 1 \cdot 7 + 3$$
$$7 = 2 \cdot 3 + 1$$
$$3 = 3 \cdot 1$$

$$1 = 7 - 2 \cdot 3 = 7 - 2 \cdot (10 - 1 \cdot 7) = 7 - 2 \cdot 10 + 2 \cdot 7 = 3 \cdot 7 - 2 \cdot 10 = 3 \cdot (17 - 1 \cdot 10) - 2 \cdot 10$$

$$= 3 \cdot 17 - 3 \cdot 10 - 2 \cdot 10 = 3 \cdot 17 - 5 \cdot 10 = 3 \cdot 17 - 5(4056 - 238 \cdot 17)$$

$$= 3 \cdot 17 - 5 \cdot 4056 + 1190 \cdot 17 = \mathbf{1193} \cdot 17 - 5 \cdot 4056$$

Siis $d = 1193$.

Salataan esimerkkinä kirjainyhdistelmä "ko". Luvuksi muutettuna sanasta saadaan luku 1014 ja salatuksi luvuksi saadaan

$$E(P) = C \equiv 1014^{17} = 3694 \pmod{4187}.$$

Salattu koodi puretaan

$$D(C) \equiv 3694^{1193} = (P^{17})^{1193} = P^{17 \cdot 1193} = P^{k\phi(n)+1}$$
$$\equiv (P^{4056})^5 P \equiv P = 1014 \pmod{4187}.$$

Liitteestä kaksi löytyy RSA -salauksen totetus C++ -kielisenä. Ohjelmalistaus on melko lyhyt, koska toteutuksessa on käytetty hyväksi valmiita ohjelmakirjastoja. Ohjelmakoodin lyhyys kuitenkin osaltaan ilmentää, kuinka kätevästi lukuteorian keinoin toteutetaan varma salausmetodi. Funktiot liittyen modulaariseen potenssiinkorotukseen ja käänteisluvun laskemiseen modulo m olisi voinut ohjelmoida itse, mutta ne haettiin lukuteorian sovelluksiin orientoituneelta www -sivustolta, jonka osoite on nähtävissä liitteessä kaksi. Pseudokoodit kyseisille funktioille löytyvät mm. tässä työssä käytetystä kirjasta [2, s. 822],[2, s. 823].

5 Viitteet

Viitteet

[1] Kenneth H. Rosen. *Elementary number theory and it's applications*, Kolmas painos: Addison-Wesley, 1993. ISBN 0-201-57889-1.

[2] Thomas H.Cormen Charles E. Leiserson Ronald L. Rivest *Introduction to algorithms* Neljästoista painos: The mit press, 1994. ISBN 0-262-03141-8

6 Liitteet

Liite 1 - Ceaserin salaus C++ -kielisenä

```
#include<iostream.h>
#include<math.h>
#include<stdio.h>
#include<stdlib.h>

void main(void) {
    // Ceasarin salaus by Aki
   char aakkoset[28] = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i',
    'j', 'k', 'l', 'm', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w',
    'x', 'y', 'z', 'å', 'ä', 'ö' };
   int vastaavatLuvut[28];

    for(int i = 0; i <28; i++)vastaavatLuvut[i] = i;
    char plain[80] = "sergei pitää kaalikeitosta";
    cout << "plaintext alussa: " << plain << "/n"
    char rplain[80];    // tila ryhmitellylle viestille
    int i = 0; //indeksi plain taulukkoon
    int k = 0; //indeksi rplain taulukkoon
    int sana = 0; // indeksi sanan pituutta varten
    // ryhmittely
    while (plain[i] != '/0') // merkki kerrallaan
    {
      if (plain[i] !=  ' ' && sana != 5){
        rplain[k] = plain[i];
         sana++; i++; k++;
      }

      else if (plain[i] ==  ' '){
        i++;
      }

      else if (sana == 5){
        rplain[k] = ' ';
        k++;
         sana = 0;
      }
  }
    rplain[k] = '/0';
    cout << "rplain : " << rplain << "/n";
    cout <<  "vastaavat luvut ryhmiteltyna" << "/n/n";

    i=0; int merkinNro;
    while (rplain[i] != '/0'){
    if(rplain[i] == ' ')
      {
        cout << "/n";
         i++;
      }
```

```
   else
   {
     merkinNro = 0;
      while(aakkoset[merkinNro] != rplain[i])
      {
         merkinNro++;
      }
      cout << merkinNro << "/t";
      i++;
   }
 }
cout << "/n";
cout <<  "vastaavat luvut ryhmiteltyna ja koodattuna" << "/n/n";
 i=0; merkinNro = 0;
 while (rplain[i] != '/0'){
 if(rplain[i] == ' '){
     cout << "/n";
      i++;
   }
   else{
     merkinNro = 0;
      while(aakkoset[merkinNro] != rplain[i]){
         merkinNro++;
      }
      cout << (merkinNro + 3 % 28) << "";
      i++;
   }
 }
cout << "/n"; cout << "Caesar koodattu viesti : " << "/n/n";

i=0;    //laskurin nollaus
merkinNro = 0;
int koodattuNro = 0;
while (rplain[i] != '/0'){
 if(rplain[i] == ' '){
     cout << " ";
      i++;
   }

   else{
     merkinNro = 0;
      while(aakkoset[merkinNro] != rplain[i]){
         merkinNro++;
      }
      koodattuNro = merkinNro + 3 % 28;
      i++;
      cout << aakkoset[koodattuNro];
   }

 }

cout << "/n /n silmukka kierrettiin " << i << " kertaa" << "/n";

cin >> keskeytys;
return;
 }
```

Liite 2 - RSA -salaus C++ -kielisenä

```
#include <NTL/config.h> #include "stdafx.h" #include <ostream.h>
#include <istream.h> #include <stdlib.h>

#include <NTL/ZZ.h> // http://shoup.net/ntl/
                    // "NTL: A Library for doing Number Theory"
                    // Kirjastosta haettu funkiot
                    // PowerMod(),InvMod() sekä
                    // ja tietotyyppi ZZ isoja kokonaislukuja
                    // varten.
NTL_CLIENT int main(int argc, char* argv[]) {

    ZZ a, e, p, q, n, c;
    cout << "anna koodattava patka a:"; cin >> a;
    cout << "anna modulus e:"; cin >> e;
    cout << "luvun n tekija p: ";cin >> p;
    cout << "luvun n tekija q :";cin >> q;

    n = p*q; cout << "n : " << n << "/n";

    c = PowerMod(a, e, n);
    cout << "salattu koodi : " << c << "/n";
    // luvun e käänteisluku (mod phi_n) lasketaan Euklideen algoritmilla.
    ZZ d, phi_n, o;
    phi_n = (p-1)*(q-1);
    d = InvMod (e, phi_n);
    cout << "d : " << d << "/n";

    o = PowerMod(c, d, n); cout << "avattu koodi : " << o;
    char lopetus[1]; cin >> lopetus;
    return 0;
}
```