Föreläsning 9

Euklides algoritm
RSA Kryptering

Euklides algoritm Används till att bestämma sgd av två tal, på ett mycket snabbare sätt än att använda primtalsfaktoriseringen av talen. Den bygger på följande

Sats Om $r = a \mod b$ (dvs $a = qb + r$, $0 \leq r < b$) så är $\text{sgd}(a, b) = \text{sgd}(b, r)$

Euklides algoritm kan skrivas i pseudokod på följande sätt.
invärde är två heltal a, b, utvärde är $\text{sgd}(a, b)$

```
sgd(a, b){
  if(a < b)
    swap(a, b) // a skall alltid vara det större av talen i början
  while(b¬ = 0){
    r := a mod b //låt r vara resten vid division av a med b
    a := b // gör sedan samma sak med b och r
    b := r // fortsätt till divisionen går jämnt upp
  }
   return a
}
```

Man gör alltså om samma procedur tills man inte längre får någon rest. Det *mindre* av de två talen när man inte får någon rest är då $\text{sgd}(a, b)$

return a i algoritmen beror på att man byt namn på a och b inne i whileloopen.

Beräkningstid för Euklides algoritm: om $a, b \leq m$ för $m \geq 8$ så görs som mest $\log_{3/2} \frac{2m}{3}$ modulo operationer.

Bevis se kursboken. Kopplat till Fibonacciföljden. Uppgift motsvarande Aug 06 uppgift 4 och Dec 05 uppgift 7 ingår ej...

Exempel: Bestäm $\text{sgd}(516, 996)$ mha Euklides algoritm.
Vi byter först plats så att

$a_0 = 996$, $b_0 = 516$ sedan ska vi dividera
$996 = 1 \cdot 516 + 480$, dvs $r_0 = 480$ omdefinierar

$a_1 = 516$, $b_1 = 480$ whileloopen fortsätter. Dividerar igen
$516 = 1 \cdot 480 + 36$, $r_1 = 36$
$a_2 = 480$, $b_2 = 36$ whileloopen fortsätter. Dividerar
$480 = 13 \cdot 36 + 12$, $r_2 = 12$

$a_3 = 36$ $b_3 = 12$ While loopen fortsätter dividerar igen
$36 = 3 \cdot 12 + 0$ $r_3 = 0$ här hade vi förstås kunnat stanna....
$a_4 = 12$ $b_4 = 0$ whileloopen stannar
$\text{sgd}(996, 516) = 12$

Sats Om a och b är heltal, minst ett av dem $\neq 0$, så finns heltal s och t så att
$\text{sgd}(a, b) = sa + tb$
Vi ger inget bevis (finns i kursboken) utan visar i stället hur man bestämmer s och t i ett exempel (det ovan).

Euklides Algoritm 'baklänges'.
$\text{sgd}(996, 516) = 12$ Ur ekvationerna ovan får vi.
$12 = 480 - 13 \cdot 36 = 480 - 13 \cdot (516 - 480) = 14 \cdot 480 - 13 \cdot 516$
$= 14 \cdot (996 - 516) - 13 \cdot 516 = 14 \cdot 996 - 27 \cdot 516$
14 och -27 var talen vi sökte.

RSA Kryptering – Rivest Shamir Adleman
Meddelande – Kryptering – Dekryptering
RSA har öppen nyckel för dekryptering. Denna kan alltså den som vill motta meddelanden göra allmän så vem som helst kan göra sitt meddelande hemligt. Men man är ensam (förhoppningsvis) om att kunna dekryptera hemliga meddelanden som ska till en själv.

I RSA kryptering görs meddelanden om till tal, genom att låta varje tecken motsvara

en siffra. Meddalenden som skickas är alltså alltid tal.

Krypteringsnyckeln består av två tal z och n. Mottagaren bestämmer dessa på följande sätt.

Först väljs två (stora) primtal p och q.

Låt $z = pq$.

Det är väldigt svårt att hitta p och q (om de är väldigt stora) även om man känner till z. Därför kan z vara öppen (offentlig)

Sedan låter man $\phi = (p-1)(q-1)$ och väljer n så att $\text{sgd}(n, \phi) = 1$.

Man kan alltså exvis låta n vara ett primtal större än det största av p och q.

z, n är vad man behöver för att kunna kryptera meddelanden. Dessa är offentliga.

För att kryptera (budskapet) a där $0 \leq a \leq z-1$ beräknar man
$c = a^n \mod z$, vilket alltså vem som helst kan göra
och skickar c till mottagaren

För att dekryptera c behöver mottagaren ha beräknat ett tal s så att
$sn \mod \phi = 1$
och $0 < s < \phi$.

Då $\text{sgd}(n, \phi) = 1$ vet vi (Sats ovan) att det finns s' och t så att
$s'n + t\phi = 1$
Vi vet också hur vi bestämmer s' och t.

Eftersom $s'n = t\phi + 1$ har vi alltså $s'n \mod \phi = 1$.
s får vi genom att välja $s = s' \mod \phi$

Mottagaren beräknar sedan $c^s \mod z$, vilket faktiskt är samma som det ursprungliga a som skickades av avsändaren.

Beviset för att det är så går dock utanför kursen

Exempel: Låt $p = 3$ och $q = 7$. Då är
$z = pq = 21$,
$\phi = (p-1)(q-1) = 2 \cdot 6 = 12$
Ska välja n så att $\text{sgd}(n, \phi) = 1$. Låt $n = 5$.

Söker sedan $0 < s < \phi = 12$ så att $ns \equiv 1 \mod 12$, dvs
$5s \equiv 1 \mod 12$ Här kan man (kanske) direkt se att $s = 5$ Annars
Euk. Alg.
$12 = 2 \cdot 5 + 2$
$5 = 2 \cdot 2 + 1$

Baklänges:
$1 = 5 - 2 \cdot 2 = 5 - 2 \cdot (12 - 2 \cdot 5) = 5 \cdot 5 - 2 \cdot 12$
Dvs $5 \cdot 5 \equiv 1 \mod 5$ alltså är $s = 5$

skicka $a = 5$
$c = a^n \mod z = 5^5 \mod 21$
Hur räknar man ut det!!!?!
$5^2 = 25 \equiv 4 \mod 21$
$5^4 = (5^2)^2 \equiv 4^2 \equiv 16 \mod 21$
$5^5 = (5^4)5 \equiv 16 \cdot 5 = 80 \equiv 17 \mod 21$
Alltså $c = 17$.

För att dekryptera får man beräkna $c^5 = 17^5 \mod 21$
$17^2 = 289 \equiv 16 \mod 21$
$17^4 = (17^2)^2 \equiv 16^2 = 256 \equiv 4 \mod 21$
$17^5 = 17^4 \cdot 17 \equiv 4 \cdot 17 = 68 \equiv 5 \mod 21$