

# Konsten att multiplicera (stora) heltal

Johan Håstad



**KTH Numerical Analysis  
and Computer Science**

18 november 2006

# Stora heltal

Mental bild:

Handmultiplikation av tal med hundratals siffror.

Datormultiplikation av tal med miljontals siffror.

Mina exempel är mycket mindre..

# Vanlig multiplikation

Vi vet hur vi beräknar  $6531 \cdot 1217$

$$\begin{array}{r}
 \phantom{0000}6531 \\
 \phantom{0000}1217 \\
 \hline
 45717 \\
 65310 \\
 130620 \\
 653100 \\
 \hline
 7948227
 \end{array}$$

## Rysk bond-multiplikation

Upprepa:

Dubblera ena talet, halvera andra (avrunda neråt).

Summera de dubblerade tal som står bredvid udda tal.

## Vilket sätt är bäst?

Hur komplicerad är metoden?

Hur mycket arbete kräver den när vi väl lärt oss den?

## Vårt huvudmått

Strunta i hur svårt det är att lära sig metoden.

Räkna antalet elementära beräkningssteg.

I vårt fall att addera eller multiplicera enstaka siffror.

## Vanlig multiplikation

Anta att varje tal har  $n$  siffor.

Varje siffra i ena tal multipliceras med varje siffra i andra:  $n^2$  multiplikationer.

Ungefär lika många additioner.

Totalt  $2n^2$ .

Notation  $O(n^2)$  betyder “någon konstant gånger  $n^2$ ”.

Fyra gånger så dyrt att multiplicera dubbelt så långa tal.

## Rysk multiplikationen

Antal halveringar är

$$\log_2 10^n \approx 3n$$

och antalet siffror i det större talet blir högst  $2n$ .

Antalet operationer blir högst  $6n^2 = O(n^2)$ .

Jämförbart med vårt vanliga sätt, 4 gånger så dyrt att multiplicera dubbelt så stora tal.

## Verklig tid

Låt oss ta stora tal, med en miljard siffror. Rejåla men får plats i datorn.

$n = 10^9$  och antal operationer med vanlig algoritm  $n^2 = 10^{18}$ .

Hur lång tid tar detta på en bra modern dator?

# Överslag

Ungefär en miljard operationer på en sekund, ger  
en miljard sekunder  
vilket är  
drygt 30 år.

## Nya tankar

$$x = x_1 10^{n/2} + x_0$$

$$y = y_1 10^{n/2} + y_0$$

$n = 4$ ,  $x = 6531$  så är  $x_1 = 65$ ,  $x_0 = 31$ .

$x_1$  är de  $n/2$  mest signifikanta siffrorna, etc

$$xy = x_1 y_1 10^n + (x_1 y_0 + x_0 y_1) 10^{n/2} + x_0 y_0$$

och vi vill räkna ut

$$x_1 y_1, x_1 y_0 + x_0 y_1, x_0 y_0$$

vilket verkar kräva 4 multiplikationer av tal med  $n/2$  siffror och lite additioner.

## Ett bättre sätt

Räkna ut

$$x_1y_1, x_0y_0, (x_0 + x_1)(y_0 + y_1)$$

och sedan använd

$$x_1y_0 + x_0y_1 = (x_0 + x_1)(y_0 + y_1) - x_1y_1 - x_0y_0.$$

Vi klarar oss med med tre multiplikationer av tal med  $n/2$  siffror (eller möjligen  $n/2 + 1$ ) och lite additioner.

## Vårt tidsexempel

$$x = 6531, x_1 = 65, x_0 = 31$$

$$y = 1217, y_1 = 12, y_0 = 17$$

$$x_1 y_1 = 65 \cdot 12 = 780$$

$$(x_0 + x_1)(y_0 + y_1) = 96 \cdot 29 = 2784$$

$$x_0 y_0 = 31 \cdot 17 = 527$$

$$x_0 y_1 + x_1 y_0 = 2784 - (780 + 527) = 1477$$

och svaret blir

$$\begin{array}{r}
 \phantom{7} \phantom{9} \phantom{4} \phantom{8} \phantom{2} \phantom{2} \phantom{7} \\
 \phantom{7} \phantom{9} \phantom{4} \phantom{8} \phantom{2} \phantom{2} \phantom{7} \\
 \phantom{7} \phantom{9} \phantom{4} \phantom{8} \phantom{2} \phantom{2} \phantom{7} \\
 1 \phantom{4} \phantom{7} \phantom{7} \\
 7 \phantom{8} \phantom{0} \\
 \hline
 7 \phantom{9} \phantom{4} \phantom{8} \phantom{2} \phantom{2} \phantom{7}
 \end{array}$$

## Nästa idé

Multiplikation av ett par av 1000000-siffriga tal reduceras till multiplikation av tre par av 500000-siffriga tal.

Gör vi dessa på vanligt sätt?

# Rekursion

Upprepa på delproblem. Vi får 9 par av multiplikationer av 250000-siffriga tal etc.

Ger  $3^j$  par av multiplikationer av  $n/2^j$ -siffriga tal.

Om  $n = 2^j$  blir det arbete ungefär  $3^j = O(n^{1.58})$ .

## Vårt exempel

$$n = 10^9 \approx 2^{30}$$

Vi får ungefär

$$3^{30} \approx 2 \cdot 10^{14} \text{ operationer.}$$

Nere i tre dygn!

# Ännu bättre

Dela upp varje tal i tre delar,

$$x = x_2 10^{2n/3} + x_1 10^{n/3} + x_0,$$

och samma med  $y$ .

Visar sig att vi kan reducera till multiplikation av 5 par av tal med  $n/3$  siffror.

$n = 3^j$  kan lösas till en kostnad av ungefär

$$5^j \approx n^{1.46}$$

ytterligare något bättre.

## Bäst kända algoritmen

Dela upp i många smådelar, använd diskreta Fourier transformen.  
Ger tid

$$O(n \log n \log \log n).$$

Löser multiplikation av tal med en miljard siffror på mellan minuter  
och timmar beroende på implementation.

## Öppen fråga

Kan vi multiplicera två  $n$ -siffriga tal med  $10n$  operationer?

Min gissning är “troligen inte” men osvuret är bäst.

# Slutord

Multiplikation är inte så enkelt som vi tror....