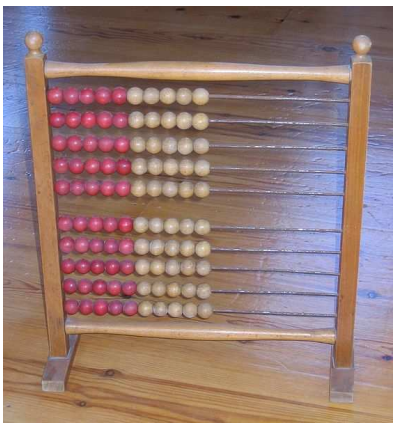
Jak násobit čísla

Představte si, že chceme zjistit, kolik je $1\,048\,576 \times 65\,536$. Můžeme na to použít třeba postup (vlastně algoritmus), který nás učili na základní škole:

$$
\begin{array}{rcccccccc}
 & & & 1 & 0 & 4 & 8 & 5 & 7 & 6 \\
\times & & & & & 6 & 5 & 5 & 3 & 6 \\
\hline
 & & & 6 & 2 & 9 & 1 & 4 & 5 & 6 \\
 & & 3 & 1 & 4 & 5 & 7 & 2 & 8 & \\
 & 5 & 2 & 4 & 2 & 8 & 8 & 0 & & \\
5 & 2 & 4 & 2 & 8 & 8 & 0 & & & \\
6 & 2 & 9 & 1 & 4 & 5 & 6 & & & \\
\hline
6 & 8 & 7 & 1 & 9 & 4 & 7 & 6 & 7 & 3 & 6
\end{array}
$$

To je zajisté správně, ale jistě uznáte, že nám to dalo spoustu práce. Pokud budeme stejným způsobem násobit dvě 100-ciferná čísla, čeká nás vyplnit 100 řádků mezivýsledků, z nichž každý má 100 číslic. Celkem tedy napíšeme 10 000 číslic, a to ještě musíme mezivýsledky sečíst. Uff ...

Obecněji, budeme-li násobit dvě n-ciferná čísla, provedeme řádově n^2 operací s číslicemi. Je jich opravdu tolik potřeba, nebo existuje nějaký rychlejší způsob?



Řešení najdete na druhé straně.

Jak násobit čísla – řešení

Rychleji to opravdu půjde. Označme si čísla, která násobíme, A a B. Pro jednoduchost předpokládejme, že obě mají stejný počet číslic n a že tento počet číslic je navíc mocnina dvojky (kdyby tomu tak nebylo, doplníme zleva nuly; tím n přinejhorším zdvojnásobíme).

Rozdělíme si číslo A na dvě části P a Q po $n/2$ číslicích, stejně tak číslo B na části R a S. Jistě bude platit:

$$A = P \cdot 10^{n/2} + Q, \quad B = R \cdot 10^{n/2} + S.$$

Hledaný součin tedy bude

$$A \cdot B = PR \cdot 10^n + (QR + PS) \cdot 10^{n/2} + QS.$$

Kdybychom znali součiny PR, QR, PS, QS (to jsou všechno součiny čísel poloviční délky), umíme z nich pár sečteními získat $A \cdot B$ – pro sečtení čísel nám přitom stačí řádově n operací, stejně tak pro násobení mocninou desítky, protože to je pouhé připisování nul na konec čísla.

Nedosti na tom, jeden ze součinů dokonce můžeme ušetřit. Stačí si všimnout, že

$$(P + Q)(R + S) = PR + PS + QR + RS.$$

Pokud tedy spočítáme $(P + Q)(R + S)$, PR a QS (což jsou tři součiny o $n/2$ cifrách), můžeme odečítáním snadno zjistit $QR + PS = (P + Q)(R + S) - PR - QS$.

Ukázali jsme tedy, že součin dvou n-ciferných čísel dokážeme spočítat pomocí tří „kratších“ součinů $n/2$-ciferných čísel a řádově n operací s číslicemi navíc. Pro ta $n/2$-ciferná čísla přitom můžeme použít stejný postup (tedy rozkládat je na $n/4$-ciferná atd.; tomu se říká rekurze), až se dostaneme k jednociferným číslům, která už umíme vynásobit přímo.

Pro počet operací, které jsme provedli, proto bude platit

$$T(n) = 3T(n/2) + cn, \quad T(1) = d$$

pro nějaké konstanty c a d. Vyřešit tuto rovnici není lehké, snadno ale můžeme dokázat, že platí

$$an^{\log_2 3} \leq T(n) \leq bn^{\log_2 3}$$

pro nějaké konstanty a, b. Jinými slovy, $T(n)$ je řádově $n^{\log_2 3} \approx n^{1{,}58}$.

Tento postup tedy pro velká n potřebuje mnohem méně operací než klasické „školní“ násobení. Není tím nejlepším známým – násobit jde ještě rychleji, ale už to nejsou algoritmy, které bychom vám uměli předvést na jedné stránce textu.