Алгоритм Кука обчислення оберненого значення.

Ділення може бути виконано так само швидко, як і множення, з точністю до сталого множника.

Щоб розділити n-бітове число *u* на *n*-бітове число *v*, можна спочатку знайти *n*-бітове наближення до числа 1/*v*, а далі помножити його на *u*, отримавши в результаті наближення до *u*/*v*.

Алгоритм Кука обчислення оберненого значення – це застосування методу Ньютона (дотичних) для наближеного розв'язання нелінійного рівняння

$\frac{1}{x} - v = 0$. Розв'язок $x = v^{-1}$ цього рівняння обчислюється наближено:

$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$, де $f(x) = \frac{1}{x} - v$, $f'(x) = -\frac{1}{x^2}$.

Таким чином послідовність: $x_{n+1} = x_n + x_n^2(1/x_n - v) = 2x_n - vx_n^2$ прямує до 1/*v*. Слід зауважити, що в останній формулі не використовується операція ділення, а тільки додавання і множення.

Алгоритм . (Обчислення оберненої величини з заданою точністю). Нехай *v* має двійкове представлення $v = (0.v_1v_2v_3...)_{(2)}$, де $v_1 = 1$. В іншому разі, або *v*=0 або $|v|$ можна представити у вигляді $|v| = 2^k \cdot v'$, де $0.5 \le v' < 1$. Тоді $v' = (0.v_1v_2v_3...)_{(2)}$, де $v_1 = 1$. Алгоритм обчислює наближення *z* числа 1/*v* таке, що $|z - 1/v| \le 2^{-n}$.

1. [Початкове наближення].
   Присвоїти $z = \frac{1}{4}\lfloor 32/(4v_1 + 2v_2 + v_3) \rfloor$ та $k = 0$.
2. [Ітерація методу Ньютона].
   (Тут маємо число z в двійковому вигляді $z = (xx.xx...x)_{(2)}$ з $2^k + 1$ знаками після коми та $z \le 2$)
   Обчислити точно $z^2 = (xxx.xx...x)_{(2)}$.
   Обчислити точно $V_k z^2$, де $V_k = (0.v_1v_2...v_{2^{k+1}+3})_{(2)}$.
   Обчислити $z = 2z - V_k z^2 + r$, де $0 \le r < 2^{-2^{k+1}-1}$ додається при необхідності заокруглення z, щоб z ставало кратним $2^{-2^{k+1}-1}$ (додавання r обнуляє знаки після $2^{k+1} + 1$ - го після коми).
   Присвоїти k=k+1.

3. [Кінець?]. Якщо $2^{k+1} < n$, то повернутися на крок 2, інакше повернути *z*.

Приклад. Обчислити 5461/43.

u=5461, v=43. Переводимо 43 в 2-ву систему числення:

v=43=(101011)(2). Тоді $v=(0.101011)_{(2)}\cdot 2^6=v'\cdot 2^6$, де $v'=(0.101011)_{(2)}$.

u=(1010101010101)(2), n=13.

Тоді $u/v=u\cdot v^{-1}=u\cdot(v')^{-1}\cdot 2^{-6}$.

$(v')^{-1}$ обчислюємо наближено за допомогою методу Кука:

k=0: $z=\frac{1}{4}\left\lfloor 32/\left(4*1+2*0+1\right)\right\rfloor=1.5=1.10_{(2)}$.

$z^2=10.0100_{(2)}$ (2.25). $V_0=0.10101_{(2)}$ (0,65625). 0,671875

$V_0z^2=1.011110100_2$. (1,4765625)

$2z-V_0z^2=2\cdot 1.10_{(2)}-1.011110100_2=11.0_2-1.011110100_2=1.100001100_2$

$2z-V_0z^2+r=1.101_2$ (заокруглення до 3-х знаків після коми).

Тобто нове значення $z=1.101_2$.(1,5234375)

k=1: $z=1.101_2$. $2^{1+1}<n=13$.

$z^2=10.101001_{(2)}$. $V_1=0.1010110_{(2)}$. $V_1z^2=1.110001100011_2$.

$2z-V_0z^2=2\cdot 1.101_{(2)}-1.110001100011_2=$

$=11.01_2-1.110001100011_2=1.011110011101_2$

$2z-V_0z^2+r=1.10000_2$ (заокруглення до 5-х знаків після коми).

Тобто нове значення $z=1.10000_2$.

k=2: $z=1.10000_2$. $2^{2+1}<n=13$.

$z^2=10.0100000000_{(2)}$. $V_1=0.1010110_{(2)}$. $V_1z^2=1.10000011000000000_2$.

$2z-V_0z^2=2\cdot 1.10000_2-1.10000011000000000_2=$

$=11.0000_2-1.10000011000000000_2=1.01111101000000000_2$

$2z-V_0z^2+r=1.011111010_2$ (заокруглення до 9-х знаків після коми).

Тобто нове значення $z=1.011111010_2$.

k=3: $z=1.011111010_2$. $2^{3+1}>n=13$. Кінець: $(v')^{-1}\approx 1.011111010_2$

$u/v=u\cdot v^{-1}=u\cdot(v')^{-1}\cdot 2^{-6}=1010101010101_2\cdot 1.011111010_2\cdot 2^{-6}=$

$=1111110111111.100000010_2\cdot 2^{-6}=1111110.111111100000010_2\approx 1111111_2=$

$=127$

Відповідь: 5461/43=127.

Множення на 2^{-5} - це просто відкидання п’яти знаків праворуч в двійковому представленні $u\cdot(v')^{-1}$. А, власне, для множення $u\cdot(v')^{-1}$ можна використати один з методів множення довгих цілих чисел.

Літ. Кнут. Искусство программирования. 2 том, стр. 344.