Modular Exponentiation

Q: Given b, n, compute $b^x \pmod n$ as efficiently as possible, as a function of x.

$b^2 = b \times b$

$b^4 = b^2 \times b^2$

$b^8 = b^4 \times b^4$ (3 multiplications)

$b^{16} = b^8 \times b^8$

$b^{32} = b^{16} \times b^{16}$

$b^3 = b^2 \times b$

$b^7 = b^4 \times b^3$

$b^{23} = b^{16} \times b^7$

$b^{55} = b^{32} \times b^{23}$

Challenges.

b^{128}

$b^2 = b \cdot b$
$b^4 = b^2 \cdot b^2$
$b^8 = b^4 \cdot b^4$
$b^{16} = b^8 \cdot b^8$
$b^{32} = b^{16} \cdot b^{16}$
$b^{64} = b^{32} \cdot b^{32}$
$b^{128} = b^{64} \cdot b^{64}$

} 7 mult's.

b^{170}

$128 + 32 = 160$
$160 + 8 = 168$
$168 + 2 = 170$

10 mults.

$170 = 128 + 32 + 8 + 2$

1 0 1 0 1 0 1 0
128 64 32 16 8 4 2 1

$$n = \sum_{i=0}^{\infty} a_i 2^i, \quad a_i \in \{0,1\}$$

b^{127}

compute modular inverse

$b^{128} \cdot b^{-1}$

$1 + 2 + 4 + 8 + 16 + 32 + 64 = 127$

1 1 1 1 1 1 1

Successive Squaring (alg. to compute $b^x \pmod n$)

① Compute b^{2^i} for all i s.t. $2^i \leq x$, by successive squaring.

$$b \xrightarrow{sq} b^2 \xrightarrow{sq} b^4 \xrightarrow{sq} b^8 \longrightarrow \dots$$

② Multiply together those which are needed to give x.

(i.e. the positions of the 1's in the binary expansion)

Successive Squaring (alg. to compute $b^x \pmod n$)

① Compute b^{2^i} for all i s.t. $2^i \leq x$, by successive squaring.

$$b \xrightarrow{sq} b^2 \xrightarrow{sq} b^4 \xrightarrow{sq} b^8 \longrightarrow \dots$$

② Multiply together those which are needed to give x.

(i.e. the positions of the 1's in the binary expansion)

Square-and-multiply AKA Double-and-add.

Exponent 25

Analyse the exponent

$25 \xrightarrow{-1} 24$ } BIC

$\downarrow \div 2$

12

$\downarrow \div 2$ } Little

6

$\downarrow \div 2$ } L

$3 \xrightarrow{-1} 2$ } B

$\downarrow \div 2$

$1 \xrightarrow{-1} 0$ } B

$\downarrow \div 2$ } B

0

B L L B B

Observation:
the BLLBB reads off binary expansion backwards!

$25 = 16 + 8 + 1$

11001 ←

Build b^n

$b^{25} \xleftarrow{\times b} b^{24}$ } B

$\uparrow sq$

b^{12}

$\uparrow sq$ } L

b^6

$\uparrow sq$ } L

$b^3 \xleftarrow{\times b} b^2$ } B

$\uparrow sq$

$b \xleftarrow{\times b} 1$

$\uparrow sq$ } B

1

How many multiplications?

Write x in binary $a_{n-1} \cdots a_0$ (n digits)

then $n = \lceil \log_2(x) \rceil$ "bitlength of x"

h = Hamming wt of binary expansion $\leq n$

= the # of 1's in " "

multiplications is $n-1 + h-1$

for either algorithm

We get a runtime

of $\sim \log(x)$ "linear"

Look ahead: