Montgomery and LogJump Reduction

$n \rightarrow$ modulus

$a = qn + r \qquad 0 \le r < |n|$

let's say we have $n = 7$
and we need to calculate
$12 * 15 \bmod 7$
$\hookrightarrow (12 \bmod 7) * (15 \bmod 7)$
$5 * 1 = 5$

$\hookrightarrow$ but the issue here is
division i.e. $12/7$ and $15/7$

$\hookrightarrow$ How can we optimise it?

Montgomery Reduction

$\downarrow$

here we convert n into a special "montgomery domain"

$\Downarrow$

in this domain reduction is performed using a different modulus R

How Montgomery Reduction works?

Setup -
- we have a modulus n
- set an auxiliary modulus R that is a power of 2 and greater than n
- calculate R^{-1} s.t. $\boxed{R.R^{-1} \equiv 1 \pmod n}$
- calculate n' : modular mul. inverse of $-n$

$$\boxed{-n.n' \equiv 1 \pmod R}$$

Conversion to "Montgomery form":

- the Montgomery form of a number a is $a' = a.R \pmod n$

Montgomery Multiplication

let's we want to multiply a and b which have montgomery form a' and b'

then the product $c' = a'.b'.R^{-1} \pmod n$

↳ typically a power of 2

↓

why?

↓

division and modulo operations with a power of 2 are extremely as we can implement through bit shifts and bitwise AND operations

↓

15 = 1111

↓

for dividing with 2, we just right shift 1 bit

↓

0111 = 7 which indeed is 15/2

↓

similarly for multiplication it's left shift a bit.

Now, this division by 'n' is slow

This is where we make use of Montgomery Reduction Algorithm (REDC)

REDC – computes results using multiplications and bit shifts, avoiding direct division by n

The REDC algorithm

given a no. $T = a' \cdot b'$

REDC(T) calculates $T \cdot R^{-1} \bmod n$ as follows:

$$m = T \pmod R \cdot n' \pmod R$$

$$t = (T + m \cdot n) / R$$

if $t \geq n$ then $t = t - n$

If $t \geq n$, then $t = t - n$

Return t

Notice the division by R which is a fast bit-shift

Conversion Back to Standard Form

after the calculations, we need to convert the final result c' to c

we do $c' \cdot R^{-1} \bmod n \rightarrow$ equivalent to $\rightarrow$ $c = \text{REDC}(c')$

Worked-out Example

Say we have $n = 13$

we choose $R = 16$ (2^4 and > 13)

n' s.t. $-13 \cdot n' \equiv 1 \pmod{16}$ $\Rightarrow$ $3 \cdot n' \equiv 1 \pmod{16}$ $\Rightarrow$ $n' = 11$ $n \cdot n' = 1 \pmod{R}$

also, $R \cdot R^{-1} \equiv 1 \pmod{n} \Rightarrow 16 \cdot R^{-1} \equiv 1 \pmod{13} \Rightarrow 3 \cdot R^{-1} \equiv 1 \pmod{13}$

$\Rightarrow R^{-1} = 9$

so, we have $n = 13$ $\quad n' = 11$

$R = 16$ $\quad R^{-1} = 9$

let's say we want to calculate $7 * 8 \bmod 13$; ideally $56 \bmod 13 = 4$

expected answer $= 4$

Step 1: convert to Montgomery form

$7' = 7 * 16 \pmod{13} = 112 \bmod 13 = 8$

$8' = 8 * 16 \pmod{13} = 128 \bmod 13 = 11$

$T = 7' \times 8' = 88$

$m = 88 \pmod{16} * 11 \pmod{16}$

$= 8$

$t = (88 + 8 * 13) / 16 = 12$

$t \geq n$? $\quad 12 \geq 13$? $\rightarrow$ NO $\Rightarrow$ Return 12

Now, we call REDC (12)

$\downarrow$

$m = 12 \pmod{16} * 11 \pmod{16} = 4$

$t = (12 + 4*13)/16 = 4$

$4 < 13 \Rightarrow$ return 4