HONORS ALGORITHMS

2024-01-12

1

input $a, b \in \mathbb{N}_0 = \{0, 1, 2, \ldots\}$

output a^b

bitlength of input $\approx \log_2 a + \log_2 b$

of output $b \cdot \log_2 a$ $\quad a \geq 2$

exponential

bit-length of n is $1 + \lfloor \log_2 n \rfloor$

$\boxed{\text{DO}}$ $= \lceil \log_2 (n+1) \rceil$

input a, b, m

output $(a^b \bmod m)$

L2

DEF <u>divisibility</u> $d \mid a$ if $(\exists x)(a = dx)$

in particular, $0 \mid 0$

<u>congruence</u>

DEF $a \equiv b \pmod{m}$ <u>if</u> $m \mid a-b$

DEF assume $m \neq 0$

$r = (a \bmod m)$ least non-neg. remainder of divion by m

DO $r = (a \bmod m) \iff \begin{cases} 0 \leq r \leq |m|-1 \\ r \equiv a \pmod{m} \end{cases}$

---

task

given $a, b, m \in \mathbb{Z}$ compute $(a^b \bmod m)$

$m \geq 1$

TASK

Given $a, b, m \in \mathbb{N}_0$, $m \geq 1$, compute $(a^b \bmod m)$

DO

Computing $(x \bmod m)$ can be done in quadratic time = # bit-operations

"long division"

---

compute a^b, reduce mod m ✗ exp. time

$b \geq 1$

```
x := 1
for k = 1 to b
    x := a x
return x
```

returns a^b

rounds $= b$: exponential in $\log b$

improved:

```
x := 1
for k = 1 to b
    x := (a x mod m)
```

we never have to deal w numbers $\geq m^2$ ($2 \log m$ bits)

REPEATED SQUARING

$(a^{32} \bmod m)$

$x := a$
for $i = 1$ to 5
  $x := (x^2 \bmod m)$

a
a^2
a^4
a^8
a^{16}
a^{32}

repeated squaring

$a^{39} =$

$a^{32} \cdot a^4 \cdot a^2 \cdot a^1$

done with $2 \cdot \log b$ modular multiplications

$\therefore$ Polynomial time

Input : a, b, m ← parameters of the algorithm they don't change during execution of alg

$a^b = a \cdot a^{b-1}$

$a^{2k} = (a^k)^2$

auxiliary Variables

A, B, X ← accumulator

initialize $A := (a \bmod m)$
$B := b$
$X := 1$

while $B \geq 1$
if B is odd then $B := B-1$, $X := (A \cdot X \bmod m)$
else $B := B/2$, $A := (A^2 \bmod m)$

return X

loop invariant : statement (Y/N) about the configuration (evaluation of variables)
the if true when we enter an execution of the loop
then true on exit

initialize $A := (a \bmod m)$
$B := b$
$X := 1$

while $B \geq 1$

if B is odd **then** $B := B-1$, $X := (A \cdot X \bmod m)$

else $B := B/2$, $A := (A^2 \bmod m)$

return X

loop invariant: statement (Y/N) about the configuration (evaluation of variables)

the if true when we enter an execution of the loop then true on exit

$$\boxed{X \cdot A^B \equiv a^b \pmod m}$$

true at beginning

→ by induction: true in each step

⟹ true at the end: $B = 0$ $\therefore$ $X \equiv a^b \pmod m$

LOOP INVARIANT

<u>loop</u>: <u>while</u> P <u>do</u> T

predicate over a set A

$\underline{P} : A \to \{0,1\}$

$\underline{P} : \mathcal{C} \to \{0,1\}$
predicate

$T : \mathcal{C} \to \mathcal{C}$
transformation

$\mathcal{C}$: configuration space

$R : \mathcal{C} \to \{0,1\}$ predicate over $\mathcal{C}$

<u>DEF</u> R is a <u>loop invariant</u> if

$$(\forall X \in \mathcal{C})(\underline{P}(X) \wedge R(X) \to R(T(X)))$$