PRIMALITY TESTING

Given n, how will you determine whether n is prime or not?

HIGH SCHOOL METHOD

For $i = 2$ to $\lceil \sqrt{n} \rceil$

If $i | n$, then return (non-prm)

End For

return (prm)

Complexity: Input length $= O(\log n)$

\# of iterations $= O(n^{1/2})$

If $m = \log_2 n$, then $2^m = n$

$2^{m/2} = n^{1/2}$

Complexity $= \cancel{\Theta(m^{1/2})}\ O(2^{m/2})$ exponential !

Fermat's Little Theorem [Recall]

Let p be a prime and let x be s.t $0 < x < p$. Then

$$x^{p-1} \equiv 1 \pmod{p}$$

Given a number n, let x be any random number in the interval $[1, n-1]$

Now let's define a Boolean function:

$$\text{Witness}(x, n) = \begin{cases} \text{TRUE}, & \text{if } x^{n-1} \not\equiv 1 \pmod{n} \\ \text{FALSE}, & \text{otherwise} \end{cases}$$

(1) If n is prime, then Witness(x, n) is FALSE. (always)

(2) If n is Composite, then Witness(x, n) is FALSE. (with probability q)

Algo (n, k)

I/P : Some odd integer $n \geq 2$

O/P : n is prime / composite

For $i = 1$ to k

$x = \text{Random}[1, n-1]$

If Witness (x, n) is TRUE, then
return(composite)

END FOR

return(prime)

- when the Algo returns Composite, then the answer is hundred per cent Correct.

- when the Algo returns Prime, then the answer is Correct with probability $(1 - q^k)$

Complexity

- The Complexity of the Algo is K times the Complexity of determining Witness(x, n).

- Before we determine the Complexity of Witness(x, n), let's look at a result.

Lemma:

- Let p be a prime s.t $p > 2$ and x be some no st $1 \leq x \leq p-1$

Now, if $x^2 \equiv 1 \pmod{p}$, then

$$\text{either } x \equiv 1 \pmod{p}$$
$$\text{or } x \equiv -1 \pmod{p}$$

eg.

$n = 7$

x	1	2	3	4	5	6
x^2	1	4	2	2	4	1

Proof: Say $x^2 \equiv 1 \pmod{p}$

$$p \mid x^2 - 1$$

$$p \mid (x-1)(x+1)$$

Either $p \mid (x-1)$ or $p \mid (x+1)$

If $p \mid (x-1)$, then $x \equiv 1 \pmod{p}$

If $p \mid (x+1)$, then $x \equiv -1 \pmod{p}$

In other words:

If n is prime, then it does not have any non-trivial square root of unity in the interval $[2, n-2]$

- Now, let's determine the complexity of Witness(x, n) which is same as that of verifying whether

$$x^{n-1} \equiv 1 \pmod{n}$$

- Write $n-1$ as $2^i \cdot m$ s.t m is odd
- x^{n-1} is equal to x^m squared i times
- Since n is odd, i must be ≥ 1.
- Find $y = x^m \bmod n$
- If $y \equiv 1 \pmod{n}$, then Witness(x, n) is FALSE
- If $y \equiv -1 \pmod{n}$, then again Witness(x, n) is FALSE
- FOR $j = 1$ to $i-1$

  $y = y^2 \bmod n$ [Note: y can't be 1 here after]

  If $y \equiv -1 \pmod{n}$, then return FALSE

  END FOR.
- RETURN TRUE

Since $n-1 = 2^i \cdot m$

$$i \leq \log n$$

$\therefore$ The complexity of the function Witness(x, n) is bounded by $\log n$

<u>Thm</u> : Given an odd no n, we can determine whether n is prime or not by performing $O(\log n)$ arithmetic ops. The answer is correct with probability $(1-q^t)$ where $(0 < q < 1)$.

This Algo is popularly called :

RABIN - MILLER PRIMALITY TESTING ALGORITHM.