

Constructing Large Primes

THEOREM (Pocklington-Lehmer theorem) Let n be odd and $n - 1 = FR$, where the complete factorization of F is known. Suppose that for every prime p dividing F there is an integer a such that $a^{n-1} \equiv 1 \pmod{n}$ and $\gcd(a^{(n-1)/p} - 1, n) = 1$. Then every prime factor of n is $\equiv 1 \pmod{F}$.

If also $F \geq \sqrt{n}$, then n is prime.

This theorem allows us to construct a new prime with about twice as many digits as the previous one.

[Doubling the size of a random prime]

Input: A prime p .

Output: A prime n near p^2 .

```
repeat {  
    let  $k$  be random between  $p/2$  and  $p$ .  
     $n = 2kp + 1$   
    if  $2^{n-1} \not\equiv 1 \pmod{n}$  restart this loop.  
    try to prove  $n$  is prime via the theorem.  
    if you succeed, end the loop.  
} until  $n$  is prime
```

By the prime number theorem, the expected number of iterations of the loop needed to find a prime n is about $\ln p$.

In applying this theorem in the algorithm above, let $F = p$ and $R = 2k$. It may seem strange to put the known factor 2 into R , but it would take longer to check the hypotheses of the theorem if we put the 2 in F . For the integer a of the theorem, try the ten primes < 30 .

To construct a large prime near X , begin with a known prime near the 2^i -th root of X , for some convenient i , and apply the algorithm i times with the known prime as the first input, and each subsequent input equal to the previous output. Adjust k in the final iteration of the loop to make the last n just the right size. The large prime p will have a rigorous proof of its primality and $p-1$ will have a large prime factor to make p immune to discovery by the Pollard $p-1$ method.