## Introduction

This paper will present the Fundamental Theorem of Arithmetic which says every positive integer can be factored as product of primes. After that we will see Fermat’s Little Theorem which gives us a test to see if a number is prime or composite. Then Carmichael numbers which are numbers satisfies Fermat’s Little Theorem but are not prime numbers. Lastly we will take a look at primality testing methods originated from Fermat’s Little Theorem.

## The Fundamental Theorem of Arithmetic

Positive integers are integers like 1,2,3,4,... and prime numbers are such numbers that can be factored into two positive integers only one way. Prime numbers are 2,3,5,7,... For example 14 is not a prime number since we can write it as 2x7 and 1x14. While 7 is a prime we can only write it as 1x7. When someone wants to study integers it is quickly seen that factoring numbers into smallest components is very useful. Factoring numbers into smallest components will be finding it is prime factors. Example 75's prime factors are 5x5x3. We can find such prime numbers for every positive integer. After decomposing some positive integers we can see that integers have only one way of factoring into primes. While it is easy to say it might be difficult to prove.

The fundamental theorem of arithmetic says that Every integer greater than 1 may be factored as a product of primes in a unique way.

#### Proof:

First we have to prove the existance of prime factors for every a > 1. Suppose by way of contradiction that there exists a integer a > 1 such that a cannot be written as a product of primes. By the Well-ordering Principle, there is a smallest such a. Then by assumption a is not prime so a = bc where 1 < b, c < a. So b and c can be written as products of prime factors (since a is the smallest positive integer than cannot be.) But since a = bc, this makes a product of prime factors, contradiction.

Second we will prove prime factorization is unique. Suppose by way of contradiction that there exists an integer a > 1 that has two different prime factorizations, say a=p1...ps=q1...qt, where the pi and qj are all primes. (We allow repetitions among the pi and qj .That way,we don't have to use exponents.) Then p1| a = q1...qt. Since p1 is prime, by the Lemma above, p1|qj for some j. Since qj is prime and p1 > 1, this means that p1 = qj. For convenience, we may renumber the qj so that p1=q1 .We can now cancel p1 from both sides of the equation above to get p2...ps = q2...qt. But p2...ps < a and by assumption a is the smallest positive integer with a non{unique prime factorization. It follows that s = t and that p2,...,ps are the same as q2,...,qt, except possibly in a different order. But since p1 = q1 as well, this is a contradition to the assumption that these were two different factorizations. Thus there cannot exist such an integer a with two different factorizations.

## Fermat's Little Theorem

Theorem: For any prime number n, and for any number a, 0 < a < n, an-1 = 1 (mod n).

Fermat’s Little Theorem has had a great influence in algorithms for primality testing which is one of the fundamental problems in algorithmic number theory. Solovay-Strassen Test, Miller-Robin Test, and AKS Test are some of primality testing algorithms based on Fermat’s Little Theorem.

Proof: 0, 1, 2, 3, 4, 5, ... , p - 1 is a list of all possible remainders. (Another way of saying this is that [0]p, [1]p, [2]p,... , [p - 1]p gives all possible congruence classes mod p. When we multiply

these numbers by a the resulting numbers 0, a, 2a, 3a, ... , (p - 1)a when reduced mod p is just a rearrangement of the original list.

The product 

For each k, 1 ≤ i ≤ p - 1, ak ≡ rk for exactly one rk, 1 ≤ rk ≤ p - 1.

a(2a)(3a) ...((p - 1)a) ≡ 1 . 2 . 3 ... (p – 1) (mod p).

Therefore, ap-1∏ ≡ (mod p).

So ap-1 ≡ 1 (mod p), since ∏, p) = 1.

## Carmichael Numbers

On October 18th, 1640, Fermat wrote in a letter to Frenicle, that whenever p is prime, p divides ap-1 - 1 for all integers a not divisible by p, a result now known as Fermat’s ‘little theorem.’ An equivalent formulation is the assertion that p divides ap – a for all integers a, whenever p is prime. The question naturally arose as to whether the primes are the only integers exceeding 1 that satisfy this criterion, but Carmichael pointed out in 1910 that 561(=3 x 11 x 17) divides a561 – a for all integers a. In 1899, Korselt had noted that one could easily test for such integers by using Korselt’s criterion.

*Korselt’s criterion:* n divides an – a for all integers a if and only if n is squarefree and p – 1 divides n – 1 for all primes p dividing n.

In a series of papers around 1910, Carmichael began an in-depth study of composite numbers with this property, which have become known as Carmichael numbers. Carmichael exhibited an algorithm to construct such numbers and stated, perhaps somewhat wishfully, that “this list (of Carmichael numbers) might be indefinitely extended.” Infinity of Carmichael numbers proven by W.R. Alford, Andrew Granville and Carl Pomerance in 1994.

Carmichael numbers are few and far between. Richard Pinch has found that there are 246,683 Carmichael numbers below 1016. Below 1016, there are 279,238,341,033,925 primes; so there is less than a one in a billion chance that a number is a Carmichael number.

## Solovay-Strassen Test

The test was propsed by Solovay and Strassen and was the first efficent algorithm for primality testing. Its starting point is a restatement of Fermat’s Little Theorem:

Theorem: For any odd prime number n, and for any number a, 0 < a < n,  = ±(mod n).

The algorithm works by selecting random integers and computing large powers of them in the ring Z/nZ, where n is the number you want to test. Also, the so called Jacobi symbol is calculated for these integers. If ever these calculations disagree, then n is composite. For if n were prime, the Jacobi symbol would in fact be the Legendre symbol, and for the Legendre symbol equality of the two methods of calculation is a theorem.

#### Algorithm:

Input n.

1. If n = mk for some k > 1 then output COMPOSITE.
2. Randomly select a, 0 < a < n .
3. If (a, n) > 1, output COMPOSITE.
4. If (a/n) = a^((n-1)/2) (mod n) then output PRIME.
5. Otherwise output COMPOSITE.

Test requires O(log n) arithmetic operations and is polynomial time.

## Miller-Rabin Test

This test was proposed by Michael Rabin slightly modifying a test by Miller. The starting point is another restatement of Fermat’s Little Theorem:

Theorem: For any odd prime n = 2s . t with t odd, and for any number a, 0 < a < n, the sequence at (mod n), at.2 (mod n), at.2^2 (mod n), ..., at.2^n (mod n) either has all 1’s or the pair -1,1 occurs somewhere in the sequence.

If n is composite, then the sequence may not satisfy the above property. Miller proved that, assuming Extended Riemann Hypothesis, for at least one a between 1 and log2n, the above sequence fails to satisfy the property when n is composite but not a prime power. Miller proved that the same hold with high probability for a random a without any hypothesis.

#### Algorithm:

Input n.

1. If n = mk for some k > 1 then output COMPOSITE.
2. Randomly select a, 0 < a < n.
3. If (a, n) > 1 output COMPOSITE.
4. Let n – 1 = 2s.t.
5. Compute the sequence at (mod n), at.2 (mod n), ..., at.2^n (mod n).
6. If The sequence is all 1’s or has a –1 followed by a 1 then output PRIME.
7. Otherwise output COMPOSITE.

Test requires O(log n) arithmetic operations and is polynomial time.

## AKS Test

This test was proposed by Agrawal, Kayal and Saxena. It is the only known deterministic polynomial time algorithm known for the problem. The starting point of this test is a slight generalization of Fermat’s Little Theorem.

Theorem: If n is prime then for any r > 0 and any a, 0 < a < n, (x + a)n = xn + a (mod n,xr – 1).

On the other hand, if n is composite and not a prime power, then it appears unlikely that the above equation holds for several a’s.

The AKS primality test, discovered in 2002, is the first provably deterministic algorithm to determine the primality of a given number with a run time which is guaranteed to be polynomail in the number of digits, thus, given a large number n, the algorithm will correctly determine whether that number is prime or not in time O(logO(1)n). Many previous primality testing algorithms existed, but they were either probabilistic in nature, had a running time slower than polynomial, or the correctness could not be guaranteed without additional hypotheses.

#### Algorithm:

Input n.

1. If n = mk for some k > 1 then output COMPOSITE.
2. Find the smallest r such that Or(n) > 4.log2n .
3. For every a, 0 < a ≤ 2 log n, do

If (a, n) > 1, output COMPOSITE.

If (x + a)n ≠ xn + s (mod n, xr – 1), output COMPOSITE.

1. Output PRIME.

## References

1. Agrawal, Manindra. "Primality Tests Based on Fermat’s Little Theorem." Indian Institute of Technology, n.d. Web.
2. Blecksmith, Richard. "Fermat's Little Theorem" Northern Illinois University, n.d. Web.
3. E. Lee Lady. "Fundamental Theorem of Arithmetic Proof." University of Hawaii, n.d. Web.
4. Granville, Andrew. "The Fundamental Theorem of Arithmetic" Université De Montréal, n.d. Web.
5. Rosenberg, Burt. "The Solovay-Strassen Primality Test." University of Miami, n.d. Web.
6. Tao, Terence. "The AKS Primality test." Web log post. N.p., n.d. Web. 27 Mar. 2013.
7. Zachary S. McGregor-Dorsey. "Methods of Primality Testing." MIT Undergraduate Journal of Mathematics, n.d. Web.