Testing for prime numbers using calculus

Elizabeth Mills and Yi Ming Yu, New York City College of Technology, Fall 2010

An integer b is divisible by another integer a, expressed as $a \mid b$, if there is an integer such that $b = ad$. Every integer greater than 1 has at least two divisors, 1 and itself. Prime numbers are those positive integers greater than 1 which have only 1 and themselves as divisors and no others.

Prime numbers are infinite

There are several excellent proofs of the infinity of prime numbers but the simplest is Euclid's. We start by imagining that the list of primes is finite and create a new integer from their product plus 1

$$P = p_1 p_2 p_3 \dots p_n \text{ and } n = P + 1$$

which must be either prime or not. If it is prime then we have a new prime not on our list; if it is not prime then it can be expressed as a factor of at least one prime, which must be on our list. Since it was on the original list the prime which divides n also divides P, and so must divide the difference between them, which is 1 (one of the properties of division above). However there is no prime for which $p \mid 1$ is true, since all primes are > 1 so our finite list must be missing at least one prime. Although they are infinite, prime numbers are still countable; being a subset of integers, which are countable, they cannot be more numerous than the integers.

Primality tests

A primality test is really testing for compositeness. Failure means a number is definitely composite, while passing just qualifies a number to stay in the running for primality. Passing multiple tests makes the case for primality even stronger.

The most elementary test for primality is "trial division". As the name implies you simply attempt to divide a number until you reach a divisor, at which point you know it is composite. To check a number n it is divided by an odd integer m. As m increases n is more likely to be prime and the test can end when $\sqrt{n}$ is reached.

If n is not prime then it has some divisors d_1 and d_2 which are $1 < d_1 < n$ and $1 < d_2 < n$. Since $n = d_1 d_2$ if they were both $> \sqrt{n}$ then $d_1 d_2 > \sqrt{n}\sqrt{n} = n$, but that can't be true. Hence at least one divisor must be $< \sqrt{n}$. Once we have found the lesser divisor it no longer matters what its partner above $\sqrt{n}$ is, primality has already been disproved.

The nth derivative test

In 2007 a strikingly simple primality test by Dennis Walsh was published. He proved that if the nth derivative of the function $g_n(x) = \sum_{k=1}^{n-1} e^{x^k/k}$ taken at $x = 0$ is 1 then n is prime. He gave the Maple code to test a number as

```
eval(diff(sum(exp((t^k)/k),k=1..n-1),t$n),t=0);
```

where n is the number being tested.

This test will actually work with any number of functions that are continuous and infinitely differentiable around $x = 0$. One of the faster functions for Maple to differentiate repeatedly is the binomial formula $\left(1+x^k\right)^m$. To see why trial differentiation works we only have to look at the series expansions of these functions. If we assign the function $f_n(x) = \sum_{k=1}^{n-1} \left(1+x^k\right)^m$ and then take n derivatives, in its expanded form we'll have

$$f_n(x) = \sum_{k=1}^{n-1} \sum_{j=0}^{m} \binom{m}{j} \left(x^k\right)^j$$

$$\frac{d^n}{dx^n} f_n(x) = \sum_{k=1}^{n-1} \sum_{j=0}^{m} \binom{m}{j} \frac{d^n}{dx^n} x^{jk}$$

$$\frac{d^n}{dx^n} f_n(x) = \sum_{k=1}^{n-1} \sum_{j=0}^{m} \binom{m}{j} (jk)(jk-1)\cdots(jk-n+1) x^{jk-n}$$

When we evaluate the result at $x = 0$ all of the terms disappear except for the case when $jk = n$, which becomes $\binom{m}{j}(n!)x^0$.

Speeding it up

We can shorten the series by skipping $k = 1$ and checking for 0 instead of 1. Since this test is actually a form of trial division, testing $k \mid n$, another easy way to truncate the sum is to only test odd integers up to $\sqrt{n}$ instead of all integers until $n-1$. So we can change the sum to

$$\sum_{k=1}^{\left\lceil \frac{\sqrt{n}}{2} \right\rceil} \left(1+x^{2k+1}\right)^m$$

Using $\left(1+x^k\right)^m$ is slightly different than e^{x^k} because its series expansion only continues to m rather than being infinite, so we have to choose an appropriate m. Since we're looking for instances where $jk = n$ we only need j to increase up to $n/\left\lceil \frac{\sqrt{n}}{2} \right\rceil$ (or roughly $2\sqrt{n}$), since any j above that number is unnecessary. So our final $f_n(x)$ will become

$$\sum_{k=1}^{\left\lceil \frac{\sqrt{n}}{2} \right\rceil} \left(1+x^{2k+1}\right)^{\lceil 2\sqrt{n} \rceil}$$

and the new Maple code will be

```
q:=ceil(sqrt(n)/2);
m:=ceil(k/n);
eval(diff(sum((1+x^(2*k+1))^m,k=1..q),x$n),x=0);
```

The terms of $\sum_{k=1}^{n-1} e^{x^k}$

For each factor of n it's the first term of a k's derivative which fluoresces at the nth derivative. Looking at the procession of e^{x^2} and e^{x^3} we can see that the beginning x^k part is reduced to only a constant over and over again as multiples of k are reached. If x^k were on it's own it would simply dissipate at the $(k+1)$th derivative but here e^x acts to reseed the equation with a new x^k each time that happens.

	e^{x^2}	e^{x^3}
f'	$2xe^{x^2}$	$3x^2e^{x^3}$
f''	$2e^{x^2}+4x^2e^{x^2}$	$6xe^{x^3}+9x^2e^{x^3}$
f'''	$0+4xe^{x^2}+\cdots$	$6e^{x^3}+18x^3e^{x^3}+\cdots$
f^4	$4e^{x^2}+8x^2e^{x^2}+\cdots$	$0+18x^2e^{x^3}+\cdots$
f^5	$0+8xe^{x^2}+\cdots$	$36xe^{x^3}+54x^4e^{x^3}+\cdots$
f^6	$8e^{x^2}+16x^2e^{x^2}+\cdots$	$36e^{x^3}+108x^3e^{x^3}+\cdots$

All other terms besides the first for each k are just an unfortunate (when the unwieldy size as n increases is considered) side effect. So we can decrease the size of the series being differentiated, and presumably increase the speed, if we delete any terms added by the product rule.

Prime numbers and cryptography

In traditional encryption models the host of a top secret dinner party would develop both the encrypting and decrypting functions, or at least the necessary keys to an exisiting algorithm. Then they would need to privately transmit those keys, along with the decrypting function if necessary, only to the invitees so they could then decipher the message. In the public key model the invitees would have a key available that anyone could use to encrypt a message only they can read.

The most popular public key cryptosystem, RSA, is based on factoring being far more difficult than identifying two large prime numbers. The user chooses two large prime numbers p and q through a random (enough) number generator. First a number is generated (and if it is even the number is replaced by itself $+1$) then it is tested for primality. If it proves to be composite rather than prime then the next odd number is tested and so on until one passes the test.

After both primes have been found and verified their product is generated $n = pq$ and then used to compute $N = (p-1)(q-1)$. A third, smaller number $1 < e < N$ is selected, where e and N are relatively prime (that is they share no common factors besides 1); e doesn't necessarily have to be prime although if it is that fulfills the requirement that it be relatively prime to N. e and n are the enciphering keys and are used to find the deciphering key $d = e^{-1} \bmod N$.

The function encoding messages ends up being $f(P) \equiv P^e \bmod n$ while the decoding function is $f^{-1}(C) \equiv C^d \bmod n$. Although e and n are publicly available it would be very difficult to factor n in order to find p and q, and subsequently N and then d.

Acknowledgements

This research is supported by NSF Grant 0622493 through the Emerging Scholars Program of New York City College of Technology, CUNY. Thank you to Dean Pamela Brown and Professor Satyanand Singh.

References

[1] Eric Bach and Jonathan Sorenson, *Sieve algorithms for perfect power testing,* Algorithmica **9** (1993), no. 4, 313–328.
[2] Underwood Dudley, *Elementary number theory*, W.H. Freeman and Company, 1969.
[3] A. Granville, *It is easy to determine whether a given integer is prime*, American Mathematical Society **42** (2004), no. 1, 3–38.
[4] Neal Koblitz, *A course in number theory and cryptography*, 2nd ed., Springer-Verlag, 1994.
[5] James J. Tattersall, *Elementary number theory in nine chapters*, Cambridge University Press, 1999.
[6] Dennis P Walsh, *A curious way to test for primes*, Mathematics Magazine **80** (2007), no. 4, 302–303.