Math 189H Joy of Numbers Activity Log

Thursday, October 27, 2011

*Felix Klein: "Everyone knows what a curve is, until he has studied enough mathematics to become confused through the countless number of possible exceptions."*

*Winston Churchill: "I like pigs. Dogs look up to us. Cats look down on us. Pigs treat us as equals."*

$353^4 = 15,527,402,881 = 30^4 + 120^4 + 272^4 + 315^4$ is the only known fourth power that is the sum of four other (positive) fourth powers.

Last time we reached a milestone of the class: we finished showing that

*For any integer $n$, if $\phi(n)$ = the number of integers between 1 and $n$ that are relatively prime to $n$, then for any $a$ with gcd$(n,a) = 1$ we have $a^{\phi(n)} \underset{n}{\equiv} 1$.*

We will spend some time finding many different uses for this result! The first comes from what this says about prime numbers:

*For an prime $p$, we have $\phi(p) = p - 1$, and so for any $a$ with gcd$(p,a) = 1$ (i.e., $a$ is not a multiple of $p$) we have $a^{p-1} \underset{p}{\equiv} 1$.*

Of course, as written, this requires us to know that $p$ is prime. But if you think about the statement backwards, it says that if $a$ is not a multiple of $p$, and $a^{p-1}$ is <u>not</u> congruent to 1 modulo $p$, then $p$ <u>cannot</u> be prime (and we should probably stop calling it $p$ ?). In other words, computing $a^{n-1}$ modulo $n$ (for $a$ relatively prime to $n$), and not getting 1 for an answer, <u>proves</u> that $n$ is <u>not</u> prime. And the big point is, it does so without finding a factor of $n$ (as a 'witness' to its compositeness)! The other big point is that in practice (and with some work we may be able to show why this is true?) this is very effective at sniffing out non-primes. That is, 'usually' if $n$ is not prime then picking any old base $a$ (like $a = 2$) we will find that $a^{n-1}$ fails to be 1 modulo $n$. We fired up Maple and tried this out with lots of randomly chosen numbers, using the base $a = 12$, and found that it was tough to randomly type in a number that wasn't already a multiple of 3, 13, or 17 (although that was partly just bad luck), and that in our limited experiments we never found a non-prime $n$ for which $12^{n-1} \underset{n}{\equiv} 1$.

Not that there aren't any; in fact, we remembered that we ran across one such computation back when we originally formulated our conjecture: $10^{90} \equiv 1 \bmod 91$. In general a non-prime $n$ for which, for some integer $a$, it is true that $a^{n-1} \underset{n}{\equiv} 1$ gets a special name; it is called a *pseudoprime to the base $a$*. So our old calculation means that 91 is a pseudoprime (sometime abbreviated '$\psi$-prime') to the base 10. We know from what we learned before that $n$ cannot be a $\psi$-prime to a base that is not relatively prime to $n$: if $\gcd(n,a) = d > 1$, then $d|a^{n-1}$ as well, so can't leave remainder 1 on division by $d$; the remainder will also be a multiple of $d$. But it actually is possible for a non-prime number $n$ to be a $\psi$-prime to <u>every</u> base $a$ that <u>is</u> relatively prime to $n$; such numbers are called *Carmichael numbers*, in honor of the first person to discover one. That first one is $561 = 3 \cdot 11 \cdot 17$; a quick check using Maple showed that it a $\psi$-prime to every base that we cared to plug in. This means that our new test for non-prime-ness (compositicity? compositeness?) cannot succeed with some numbers, unless you hit upon a base $a$ that is a multiple of a factor of $n$ (in which case you have actually found a factor of $n$, namely $\gcd(n,a)$). Luckily though, Carmichael numbers appear to be rare; although not too rare: there is a result from the 1960's or 1970's that states that for any $n$ large enough (the notation $n >> 1$ is used as shorthand for this) there are at least $n^{2/7}$ Carmichael numbers between 1 and $n$. Whuch sounds like a lot! until you think in terms of 100-digit numbers or so; it basically says that there are about $10^{43}$ Camichael numbers below $10^{100}$, which means that if you pick a 100-digit number a random there is only a 1 in $10^{57}$ chance of it being a Carmichael number. Which is really, really, really, small....

There is actually a characterization of Charmicael numbers $n$, if you happen to know its prime factorization; $n$ is a Charmicael number if and only if it is a product $n = p_1 \cdots p_k$ of distinct primes $p_1 < \cdots < p_k$, and if for each prime factor $p_i$ we have $p_i - 1 | n - 1$. Which, it turns out, it is tough to arrange! The second smallest Carmichael number is 1105 (as one of you thoughtfully reported); $1105 = 5 \cdot 13 \cdot 17$, and we could verify directly that $4, 12$ and 16 all divide 1104.

But this is dwelling on the failures of our new test to detect non-primes! The real point is that it is usually quite good at determining that a number is not prime. Which is why when $a^{n-1} \underset{n}{\equiv} 1$ (and if, we are suspicious, we can test several more bases as well) holds for several choices of $a$ we can be pretty confident that we have managed to discover a prime number. [This is, in fact, (almost) exactly what the computers in your instructor's basement are doing to find large prime numbers; they find likely candidates by carrying out precisely these computations.]

But by dwelling on the results of these computations, we almost missed an important fact! We were happily typing in some pretty big numbers $n$ (up to around 350 digits?), and telling Maple to to compute the remainder that $12^{n-1}$ left on division by $n$. But $12^{n-1}$ is an incredibly huge number! Even if you stick to 'tiny' numbers $n$, with maybe 15 digits (that's what, the hundreds of trillions?), $12^{n-1}$ will have hundreds of trillions of digits! And in fact if you type '12^n;' into Maple, it will refuse to compute it, saying 'Error, numeric exception: overflow'. The number is just too big. [We actually typed '12&^$(n-1)$ mod $n$'... See below!] [Right around here we had a little discussion about the origin of the word 'ampersand', as 'and, per se, and', which might or might not actually be true, but has no bearing on our further discussion...] But the point is that we never really want to know this number, just it's remainder on division by $n$. And because we know that the remainder of $ab$ mod $n$ can be computed by first computing the remainders of $a$ and $b$, multiplying them together, and then taking the remainder of that product, we can carry out an exponentiation $a^{n-1}$, mod $n$, as repeated multiplication, without ever multiplying together numbers bigger than $n$.

But this still misses part of the point. Repeatedly multiplying by $a$ (and reducing mod $n$) still takes too long, if your are going to multiply $n-1$ $a$'s together. For a 15-digit number $n$ (which, remember, we still think of as 'small'), even if you could do a billion muliplications per second, it would still take you a million seconds (around 12 days) to finish. And we were dealing with much larger numbers. But the calculations took Maple almost no time (we noted that after many such calculations it reported having used 1/10th of a second of cpu time). Clearly it is being smarter than we were about what it was doing?

The point is that our 'serial' view of exponentiation, $a^k = a \cdot a^{k-1}$, multiplying by $a$ one at a time, takes, well, $k-1$ multiplications. But if you try to figure out how to re-use your work, you can significantly speed up the process. How? Well, basically, our serial view splits the exponent, $k$, into the two pieces 1 and $k-1$, takes the result of raising $a$ to those two powers, and multiplies the results together. But if you split $k$ into diferent pieces, you can use the computation of one to help you do the computation of the other <u>faster</u>. The idea is exemplified by considering $k = 2\ell$; writing this as $k = \ell + \ell$ then $a^k = a^\ell \cdot a^\ell$, and by doing effectively half as much work, to compute $a^\ell \pmod{n}$, and squaring (which is just one more multiplcation), we have computed $a^k$ in half the time. Of course $k$ needn't be even, but it is either $2\ell$ or $2\ell+1$, so we either square $a^\ell$ or we square $a^\ell$ and multiply by $a$ (which is just one more multiplication) to compute $a^k$.

Of course, this still leaves the question of how to compute $a^\ell$ modulo $n$. But we realized that if doing this once replaced our work with half as much work, plus 2, then doing it again would be even better! The basic idea came down to:

To compute $a^k$ mod $n$, write $k_1 = \lfloor k/2 \rfloor$ (where, you may remember, $\lfloor x \rfloor$ = the 'floor' function, the largest integer $\leq x$), so $k = 2k_1 + \epsilon_1$, where $\epsilon_1 = 0$ or 1, and then knowing $a^{k_1}$ mod $n$ will allow us to (quickly!) compute $a^k$ mod $n$. But then we just continue; $k_2 = \lfloor k_1/2 \rfloor$, so $k_1 = 2k_2 + \epsilon_2$. Eventually we reach $k_r = 0$ (so $k_{r-1} = 0$ or 1), and then by working our way back up the list, we can compute

$a^{k_i} \underset{n}{\equiv} A_{i+1}A_{i+1}a^{\epsilon_{i+1}}$, where $a^{k_{i+1}} \underset{n}{\equiv} A_{i+1}$,

which requires only one (or two) multiplications modulo $n$ for each turn of the crank. How many such multiplications will we do? Each $k_{i+1}$ is at most half of $k_i$, so $k_i \leq k/2^i$ [why? induction!]. So when $2^r \geq k$ we will have $k_r \leq 1$ and be done.

Illustrating this with an example, we first tried this with $k = 15,222,911$, but then got bored with it and started over with $k = 23$; we then have $23 = 2 \cdot 11 + 1$, $11 = 2 \cdot 5 + 1$, $5 = 2 \cdot 2 + 1$, $2 = 2 \cdot 1 + 0$. So to compute $a^{23} \pmod{n}$, we compute $a$, $a^2 = a \cdot a = b$, $a^5 = b^2 \cdot a = c$, $a^{11} = c^2 \cdot a = d$, and then $a^{23} = d^2 \cdot a$. all modulo $n$, of course! And this computation would require 7 muliplications modulo $n$, not $22 = 23 - 1$. Which may not sound like much, but doing this for a number $k$ in the range of $10^{15}$ (using the useful rule of thumb that $2^{10} = 1024 \approx 10^3$) requires about 50 divisions by two, so computing $a^k$ mod $n$ in this way will require 50 to 100 multiplications mod $n$. Which is much better than $10^{15}$ of them! This is, in fact, exactly how Maple and other mathematical software (if they are well-written!) carry out such calculations, and it is exactly what the Maple command '$a$&^$k$ mod $n$' will do.

With this methodology, computing $a^k$ mod $n$ becomes <u>very</u> fast, even for 'economically' large numbers, making for a very effective test for compositeness, and many other things besides!