Subject: Re: Large Numbers
Posted by Chris[6] on Sat, 07 Feb 2009 00:49:16 GMT
View Forum Message <> Reply to Message

On Feb 6, 1:12 pm, David Fanning <n...@dfanning.com> wrote:
> Folks,
>
> I made a big mistake and signed up for an Applied Statistics
> class this semester. Now I pretty much spend every free
> waking moment doing stats homework. :-(
>
> Anyway, for lunch today I decided to grab a sandwich and
> give my youngest some support by calculating how many
> girls he had to ask out to have an 80% chance of getting
> a date for Saturday night.
>
> I made some conservative assumptions (I learned later
> my ideas about the college social scene apply more to the
> 1970s than they do to today), and off I went writing a
> couple of short IDL programs to do the calculations for
> the Binomial and Geometry Distributions, etc. All pretty
> straightforward.
>
> But then I started getting screwy results. (This, in itself,
> is not all that unusual in this particular class. In fact, I've
> begun to consider it something of a minor miracle if I'm within
> an order of magnitude of the right answer.) But even I know
> that negative probabilities don't show up until the second
> semester. What in the world!?
>
> It turns out that the recursive function I naively wrote to
> process a factorial calculation was overflowing my long
> integers, even with a simple calculation like 20! (twenty
> factorial). Yowser!
>
> Now, of course, the formula I was using has a large
> factorial number divided by another large factorial
> number, so the *actual* number I wanted to use in the
> calculation is not that big. But it begs the question:
> what strategy do computer scientists use to deal with
> one very, very big number divided by another very, very
> big number?
>
> I've solved my immediate problem for my little toy problem
> by using LONG64 variables. But this can't be the right solution.
> Does anyone know?
>
> Cheers,

>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.dfanning.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Java, for example, has a bigInteger class, which internally represents
a big integer as an array of 32 bit integers - something like
decimal_equivalent = sum(array[i] * (2^32)^i)

http://developer.classpath.org/doc/java/math/BigInteger-sour ce.html

chris
