Notes on Bernoulli Numbers for Evan

Dr. J

May 8, 2020

1 Note on subscripts

(This section was originally titled 'Quick Note on Subscripts,' but that name has become increasingly inaccurate, so I've changed it. You can see the progression in the git commits.)

Bernoulli numbers have been subscripted in three different ways:

1. Where $B_1 = -\frac{1}{2}$. This is now standard (?), and is what I use in my dissertation. You probably learned it this way. If not, let me know. That's what I'll use unless I hear otherwise from you.

2. Where $B_1 = \frac{1}{2}$. This was common in algebraic number theory pre-1980, e.g. Iwasawa or Washington. B_n for $n \neq 1$ agrees entirely with 1., above, so it only differs in this one location.

3. Where $B_1 = \frac{1}{6}$. In this case, B_n of this kind $= B_{2n}$ of either 1. or 2., above. This was common in Algebraic Topology, e.g. Milnor (one of the true giants of 20^{th}-century mathematics, still alive and well in NJ as far as I know) and Lance (my advisor, who you see mentioned several times in my dissertation).

In a line of the Preface that made me laugh out loud when I read it (though I doubt it will for you), Introduction to Cyclotomic Fields, by Larry Washington of University of Maryland said:"At Serge Lang's urging I have let the first Bernoulli number be $B_1 = -\frac{1}{2}$ rather than $+\frac{1}{2}$. This disagrees with Iwasawa [Washington's advisor at Princeton] and several of my papers, but conforms to what is becoming standard usage." Serge Lang was well-known for churning out huge textbooks in almost any field of graduate-level mathematics, whether he was an expert in that field or not. (This is likely connected to Lang's membership in the Bourbaki, which, if you don't know the story of Nicholas Bourbaki, you should look it up or ask me). So of course Lang would have done this. Lang also famously traveled with a delegation to the Republic of South Africa where many thousands of people were dying in an AIDS epidemic; they successfully convinced the government there that the HIV virus did not cause AIDS, and that preventing transmission of HIV would not slow the epidemic. This was disastrous, and the policies of the RSA government following this resulted in much loss of life there.

Just banging around doing a little research, it seems that this is not so fully resolved/ standard as I had thought, and is still in debate: http://luschny.de/math/zeta/The-Bernoulli-Manifesto.html: link to Conversation between Peter Luschny and Donald Knuth on this topic. Note that Donald Knuth is the same guy who, back in the 1970's, got so fed up with trying to type a mathematical paper for submission to a journal that he decided to create the first version of $\TeX$ so that he could properly typeset it.

I think it is good for you as a young mathematician, to look at a debate on the definition of something (in this case, the Bernoulli numbers), and see what goes into it. When you study mathematics, typically, you just have a definition in a book, and it's sitting there, complete and perfect, and you memorize and then you learn what it really means, and then you learn to use it (which are all different things). But, there was a long road to get to the point where it was decided what should be in that definition and what should not be in that definition. And there are other formulations that are logically equivalent, and therefore, are in some sense, no different, but that lead mathematicians in other directions, or are more complex to understand, or are harder to learn to use, and so have been rejected. There is psychology in the mathematics in this case – we choose the version that helps people learn the mathematics, or the one that is most elegant, or the one that makes the most connections. Often, this is not a controversial process, and mathematicians settle pretty quickly on something, but sometimes this conversation serves a guide to the entire branch of mathematics, its development and its history. The terms "regular prime/irregular prime" have this to some extent, and I talk really briefly about this in my dissertation also.

2 A Computational Technique

We discussed this in our meeting in class today, but I wanted to write it up carefully and add a bit to it, to give some context and give you some things to think about.

Example 1: This is example I rattled off at our meeting: Calculate $2^{2020} \pmod{11}$.

We know that $2^5 = 32 \equiv -1 \pmod{11}$. So, $2^{10} = \left(2^5\right)^2 \equiv (-1)^2 = 1 \pmod{11}$. Then, $2^{2020} = \left(2^{10}\right)^{202} \equiv (1)^{202} = 1 \pmod{11}$.

Or, if you know Fermat's Little Theorem, we can apply that, which says (for prime modulus), that $a^{p-1} \equiv 1 \pmod{p}$, for all $a, 1 \leq a \leq p-1$. In general, it says that $a^{\phi(n)} \equiv 1 \pmod{n}$ for all a coprime to n (where ϕ is the Euler ϕ-function, not sure if you know this or not). Notice it does *not* say that this is the smallest power of a that is congruent to 1, for any particular a. It is, however, not hard to show that this is the smallest power that works for all coprime a.

Finding solutions to $a^n \equiv b \pmod{p}$ (so, in effect, finding n^{th} roots $\pmod{p}$) is an interesting and deep question with few obvious answers (the study of *primitive roots*).

Notice that, in Example 1, we never had to use any number with more than two digits. If you did the obvious thing in your computer, and asked it to compute `2**2020 % 11`, it can do that, but it's a lot of computational work. It first has to compute all of 2^{2020}, and then reduce that mod 11. To see what that looks like: (commas omitted), $2^{2020} =$

```
120 390 229 192 789 671 200 196 730 675 808 906 407 818 580 678 535
565 853 604 471 040 981 468 330 576 609 422 256 057 752 381 687 848
600 439 581 729 091 776 513 008 621 150 593 910 720 527 739 772 380
453 052 486 767 498 034 969 314 002 237 284 144 953 291 103 458 547
532 810 152 608 127 216 408 475 325 114 421 897 897 408 047 581 395
677 670 971 695 493 487 923 933 346 069 636 224 032 935 216 763 561
673 143 257 907 287 561 970 520 670 661 943 292 226 106 584 203 713
841 952 673 366 886 865 445 199 267 790 891 789 863 232 017 223 226
748 196 794 533 959 989 836 805 876 911 810 211 481 167 739 679 043
319 937 687 835 412 885 323 948 134 322 098 370 385 629 943 305 785
136 881 090 458 653 857 068 542 385 988 740 344 220 360 507 575 957
485 047 851 613 181 253 218 943 644 136 742 478 444 626 968 576
```

and 2020 is a really small power for what we're looking at.

Another (better) way to look at the computational cost is that the number of digits in 2^{2020} is equal to $\log_{10}(2^{2020}) = 2020 \cdot \log_{10}(2) \sim 2020 \cdot (0.3013) \sim 608.6$, so 2^{2020} should have 609 digits as a base 10 number (which you can verify above if you're bored).

In general, if you are computing powers of integers $\pmod{n}$, your computation should never need to use a number bigger than $\pmod{n^2}$. This is a *huge* computational advantage, and, using $\log_2$, you can figure out how many bits that takes up in your computer. Keeping it under 64 bits is a *huge* deal because it allows you to do the computation all at once in the processor (most modern processors are 64-bit processors).

Example 2: Compute $2^{2020} \pmod{37}$. By Fermat, we know that $2^{36} \equiv 1 \pmod{37}$. So, $2^m \equiv 2^n \pmod{37}$ if $m \equiv n \pmod{36}$. So, I'll reduce 2020 (mod 36); to keep that in my head, I'll use easy multiples of 36: $2020 - 1800 = 220$; $220 - 180 = 40$, $40 - 36 = 4$, so $2020 \equiv 4 \pmod{36}$. This tells us that $2^{2020} \equiv 2^4 \pmod{37}$. So $2^{2020} \equiv 16 \pmod{37}$.

Exercises:

1. Compute (by hand):

   (a) $2^{2020} \pmod{17}$

   (b) $5^{2020} \pmod{17}$

   (c) $6^{31415} \pmod{23}$

2. Find an example of a, n, p where $a^n \equiv 1 \pmod{p}$, $a > 1$ and $n < p - 1$. This is a counterexample to show that Fermat's Little Theorem does not give the minimal exponent for all a. Obviously, if $a = 1$ this is trivial. (Hint: If you are looking for cases where $a^n \equiv 1 \pmod{p}$, you should be looking at values of n that divide $p - 1$. Why is that?)

3. Show that, for a in $1 \leq a \leq p - 1$ and p prime, $a^n \not\equiv 0 \pmod{p}$ for all positive integers n.

4. Find an example where $1 \leq a \leq m-1$, m composite, and $a^n \equiv 0 \pmod{m}$ for some a, m, n.

5. On paper, write down a process that, for any $a, 1 < a \leq p-1$, a positive integer k, and prime p, will allow you to calculate $a^k \pmod{p}$. Use Fermat's Little Theorem.

6. Write a function in Python that returns the value of $a^k \pmod{p}$ for any a, k, p where $1 < a \leq p-1$, n a positive integer, and p prime. Use the method that we developed above. You do not need to validate the inputs against these requirements, but you can assume that they are true in the design of your algorithm.

7. Explain how $\log_2(n)$ can be used to tell us how many bits an unsigned integer will occupy in your computer. What happens if n is allowed to be signed? What if n has a decimal part? (If you don't actually know the answers to these, that is ok, think about it and we can talk about next time we meet. Don't feel like you need to go research this.)