Gimme a date! John Conway's shortcuts to cracking the calendar

Stephen D. Miller*
Yeshiva University
stephen.miller@yu.edu

September 16, 2026

1 A magical visit

It's such a pleasure to recall a premier educational experience, a happenstance visit to John Conway's office around 8pm one weekday evening in the spring of 1994. Like any other first-year Ph.D. student at Princeton, I had already of course seen Conway's gatekeeper to the Sun workstation in his office: upon login, his `gad` program[1] forced him to enter the day of the week of ten randomly chosen dates before accessing his computer. The program then informed him the time it took him to pass this drill, and he in turn recorded his record-setting times on a sign on top of his monitor, conspicuously displaying them to anyone who walked down the Princeton math department's main corridor. A photo of him smiling in front of this prominent sign ("15.92 seconds") even appeared on the front page of the New York Times Science section in 1993 [1].[2]

*Supported by NSF grant CNS-2124692. I wish to thank Adina Miller, Siobhan Roberts, and Dierk Schleicher for their helpful conversations.

[1]This program was written for Conway by friends of his outside the university, and its name is an acronym for "gimme a date".

[2]Conway was proud of the fact that even the photographer was himself famous: the legendary Cambodian journalist Dith Pran, whose portrayal by Haing S. Ngor in *The Killing Fields* won an Oscar for Best Supporting Actor.

Curiosity possessed me and for some unknown reason (not having taken this opportunity myriad other times), I decided to ask him how he manages to compute days of the week so rapidly. There was nothing Conway liked more – not even making up or practicing his own magic – than conveying this magic to someone else, and (likely due to the fact he did not know me well) he displayed a calm confidence that he could teach me. Knowing myself, I was far less confident, but all I was hoping for was to glimpse the inner workings of a legend.

What followed had the hallmarks of a transcendental piece of education: it felt magical while it was happening, seeing all the gears that were revealed as he removed the cover to his masterpiece – yet immediately afterwards everything was so clear that in retrospect that I felt I must have known it all along.[3] The material was taught in a way to be unforgettable. I don't know how he did it, but he calibrated the exact right way to convey his ideas to me based on what my responses were. The best way I can describe this is to imagine yourself as a clumsy Luke Skywalker, quickly gaining the ability to "use the force" from Obi-wan Kenobi, or to mentally levitate objects from Yoda. It felt utterly preposterous to have been unexpectedly handed the keys to the magic – what a gift! He worked me through a sequence of examples, and by the time I left his office 15 minutes later I felt like a completely different person: not because of the math or the trick itself, but from realizing how much a spectacular educator can build up the confidence of his students.

Conway's generosity did not end with this time alone: he also let me `ftp`[4] out the `gad` program so that I too could practice with it. We certainly had an interesting and eventful time playing his game together over the course of the next few weeks (as documented in [2, 3]). He was even more generous in explaining his tricks of the trade. I recognize that, for various reasons, I was fortunate to gain access to Conway's "trade secrets", and my goal in this piece is to share them with those who were not in a position to benefit from such an opportunity.

[3]On a different occasion, Conway told me that he was exceptionally satisfied by the reaction he overheard from graduate students at the University of Michigan to a lecture he gave on a theoretical simplification he had discovered about Coxeter groups: "well, that was pretty trivial." Conway relished that it *wasn't* trivial before he made it trivial.

[4]The is the old Unix file transfer protocol, since superseded by the secure `scp`.

2 Conway's Doomsday algorithm

Let's recall the basic structure of the Doomsday algorithm to compute the day of the week a given date falls on. Conway layers it in the following three levels.

Level 1 – within a year: Here one starts with the observation that in any given year, the dates

$$\begin{gathered} 4/4,\ 6/6,\ 8/8,\ 10/10,\ 12/12, \\ 9/5,\ 5/9,\ 7/11,\ 11/7, \\ 3/7,\ 2/7^{\star},\ 1/31^{\star} \end{gathered} \tag{2.1}$$

all fall on the same day of the week as each other. Conway calls that day the "Doomsday" for the year; for example, in 2026 it is Saturday, in 2027 it is Sunday, and in 2028 it jumps to Tuesday.

Here I've used the American convention of the month/day format, though the first two lines happen to work equally well in the day/month format. Conway was proud that he reduced the need to translate between these conventions through his choices in (2.1). His mnemonic for the second line was "My 9-to-5 job at the 7-11". The final line in (2.1) does not fit as nicely with the others, which makes us appreciate the elegance of the first two lines. It might be that 3/14 ("π day") is a more memorable Doomsday for March, but this is not how Conway taught it. Instead, this master teacher would initiate the lessons on dates from April through December until his students became comfortable enough to branch out to January, February, and March. The asterisks on the last two dates indicate that in leap years the Doomsday is actually the day after these listed days. In other words, (2.1) is always in place for the last 10 months of the year, but the first two months follow a different pattern in leap years.[5] As an example, January 31st falls on the Doomsday in non-leap years, but on the day before the Doomsday in a leap year.

The day of the week of any date can be recovered from (2.1) by cross-referencing it with the listed day for that month. As an example, Conway's birthday of December 26 always falls on a Doomsday, since it is exactly two weeks after December 12.

[5] The asterisk would have been unnecessary had the leap day been placed at the very end of the calendar year, as it was in many ancient calendar systems.

Level 2 – within a century:

The Doomsday for the year 2000 was a Tuesday. What is it for the year 20xy? Let's call the last two digits "xy" the integer z, which ranges in possible values from 0 to 99. Since the year is normally $365 = 52 \times 7 + 1$ days long, the Doomsday advances one day later in the week each non-leap year – and in leap years, two. Accordingly, the Doomsday in the year 20xy is

$$z \ + \ (\text{number of times 4 divides into } z) \ = \ 4 + \lfloor z/4 \rfloor \tag{2.2}$$

days later than Tuesday. Since a week has 7 days, it of course suffices to find the remainder of (2.2) after division by 7. However, this is cumbersome in practice, especially when z is large.

Conway came up with a faster way, by breaking up z a dozen years at a time. The rationale is that if $z = 12$, then the quantity in (2.2) is $12 + 3 = 15 = 2 \times 7 + 1$, so that the Doomsday moves one day forward every 12 years. Conway divides z by 12 to get a quotient q and remainder r:

$$z = 12 \times q + r.$$

To account for the number of leap years in these r leftover years, he then counts the number of times that 4 divides r, call it $s = \lfloor r/4 \rfloor$. His formula is then that the Doomsday in 20xy falls

$$q + r + s \tag{2.3}$$

days after Tuesday. For example, in 2026, 12 goes into 26 $q = 2$ times with remainder $r = 2$; the number of 4's that divide into r is $s = 0$. Indeed, the Doomsday for 2026 is Saturday, $4 = q + r + s$ days after Tuesday.

Level 3 – a date in any century:

The Gregorian Calendar repeats every 400 years, since the number of days in 400 years is a multiple of 7. Consequently, a short table suffices to tell you Doomsdays at the start of each century (see Table 1).

In the epoch of the 20th century, Conway would remind his audience that in "We-in-this-day", using the pun to help them remember the Doomsday for 1900 was a Wednesday. Conway further treated the days themselves as numbers, starting with Sunday="Noneday" = 0, Monday="Oneday" = 1, Tuesday="Twosday" = 2, ..., Thursday="Foursday" = 4, Friday="Fivesday" = 5, and Saturday="Sixesday" = 6, a surprising pattern which fits the English language magically well (...Wednesday aside).

Year	**Day**
2100	Sunday
2000	Tuesday
1900	Wednesday
1800	Friday
1700	Sunday
1600	Tuesday
1500	Wednesday

Table 1: Doomsdays in Gregorian calendar years that are multiples of 100.

For dates on the Julian calendar (which was used recently enough that my late Rutgers colleague Israel Gelfand spent the first few years of his life in Tsarist Russia under it!), the table is simpler (see Table 2).

Year	**Year**	**Day**
700	1400	Sunday
600	1300	Monday
500	1200	Tuesday
400	1100	Wednesday
300	1000	Thursday
200	900	Friday
100	800	Saturday

Table 2: Doomsdays in Julian calendar years that are multiples of 100.

Note that the drop of one day per century also happens between the 1900s and 2000s, because 2000 was a leap year and the Gregorian Calendar's innovation was to eliminate the leap day in years which are multiples of 100 but not of 400. The latter also explains the drop of two days between the 1800s and 1900s, etc..

3 How you can speed this up: time/memory tradeoff

There's a difference between calculations that are easy to learn, and calculations that are *fast* to do. The Doomsday algorithm Conway presented has essentially five numbers to add up (and then take the remainder of after division by 7): the three numbers q, r, s; the offset for the century; and of course the number of days the date in question differs from a Doomsday.

Do less math:

What I just described wasn't the method Conway himself practiced, however. Even for a speedster like Conway, adding five numbers together is slow, and computing some of them is tedious in the first place. Accordingly, a major trick to speed up the computation is to separately obtain $q + r + s$ together as one number, which I'll refer to as the "year number". With the year number in hand, only two further additions are needed. I'm not sure if this is how Conway thought of it, but memorizing the table of possibilities of adding three numbers modulo 7 is not so hard: there are only 84 possibilities, just slightly more than the 78 pairs one has to memorize for multiplication tables up to 12 (and the numbers are smaller, too).

Landmark years with easy Doomsdays:

Accordingly, Conway's bag of calendar tricks – the reason for his amazing speed – consisted of tools to quickly determine the year number. Figure 1 displays Conway's mental organization in his own pen. He lines up the years to take advantage of the periodicity every 28 years within a century, and circles those for which the year number $q + r + s$ is zero or a multiple of 7. He memorized those numbers (00, 06, 17, 23, 28, etc.), which served as his navigational landmarks. In the left margin he lists the offset of each year from zero, sometimes written as a negative number when helpful. He did not need to memorize the marginal numbers, however, because they can be quickly recovered from a nearby circled number. However, leap years throw this off to some extent, and one of the most difficult parts of the memorization approach is in the middle of the page, where he uses fractional notation to show how to split the difference between two consecutive years. For example, he adds the year $11\frac{1}{2}$ to his memorization list, to indicate that the year number for 11 is -1, whereas the year number for 12 is 1.

To be even faster, one can memorize a table similar to Figure 1, but specialized for a particular century, such as the 1900s. For example, Conway's birth year of 1937 has Doomsday Sunday (he was born on the last Doomsday of that year). Conway's `gad` program was devised to sample 40% of its challenge problems from dates in the 20th and 21st centuries; respective dates

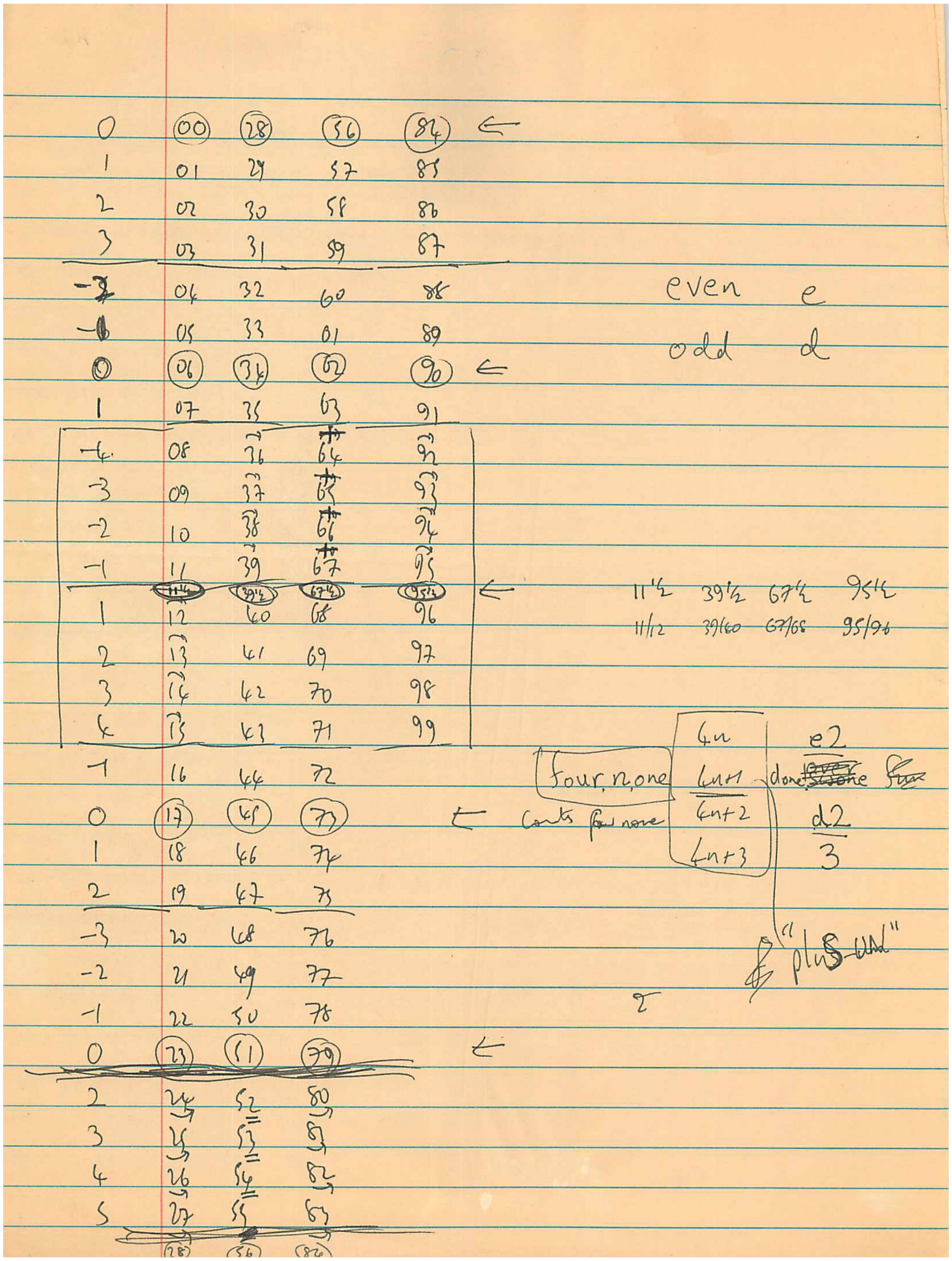

Figure 1: Conway's explanation of how to compute the year offset (Princeton, 1994). The page is barely long enough to accommodate the 28-year calendar periodicity.

between those centuries are generally adjacent, and so conversion between the two is simple. With such a table (or at least its landmarks) memorized, calculation of the Doomsday for years we've lived in becomes a simple game of recall, with only one addition needed.

In my experience teaching the Doomsday algorithm and its enhancements, the biggest mistake people make is not taking into account the subtlety that January and February dates in a leap year must be adjusted. Conway liked to work with "January 32nd" as the Doomsday for January leap years. Alternatively, one can attach January and February to the previous year.

Conway's poetic sense is on display here in his chart ("four, none, $4n+1$, done"), along with his characteristic adoption of compact notation via various other symbols. His more serious mathematical pursuits, e.g., on finite simple groups and the Leech lattice, bear the same fingerprints. Indeed, Conway was the first to explore the depths of a number of mathematical objects, and as a result imprinted his notation on them. At times it is all too easy to misidentify the inherent nature of these objects as colorful idiosyncrasies he imposed on them through the sheer will of his personality. "Sunday, Monday, Tuesday" and "Thursday, Friday" are not linguistically related to "Noneday, Oneday, Twosday" and "Foursday, Fivesday", but one could be forgiven for falling into the temporary illusion that Conway's charisma willed them to be so.

Clearly John Conway was a master of finding simple explanations of complicated processes.

4 Epilogue: Conway and the Hebrew calendar

Conway's imagination served as an advanced scout for many of us in mathematics: he went on missions to analyze complicated systems, and reported back elegant perspectives to us. He was a totally unique thinker and understood how to include competitiveness in effective teaching. I'll mention something we worked on together, which he later pursued to completion in [4]: converting between the Hebrew and Roman calendars.

To keep the discussion simple (and since [4] is so thorough), I'll focus on a simple yet crucial aspect of the Hebrew calendar, namely its pattern of leap months. The Hebrew calendar operates with lunar months, which are slightly longer than 29.5 days on average. A dozen such months consists of 354 days, which is significantly shorter than the solar year. At the same time, the Torah commands that the holiday of Passover must fall after the vernal equinox. Thus the Hebrew calendar is a hybrid lunar-solar calendar. To keep the holidays from sliding out of season, leap months are periodically added: seven out of every 19 Hebrew years have 13 months, with the additional month in the late winter (roughly around the same time the Roman calendar has its leap day, in fact).

The pattern of which years are Jewish leap years can be stated very cleanly using the Roman calendar. Since Hebrew years start a few months before Roman years, we will state things in terms of the Roman year y in which the added, 13th month falls. Indeed, most of the Hebrew year lies in this Roman year, although the Hebrew year actually starts in the Roman year $y-1$.

Leap year formula: If the year y is divided by 19 and written as $y = 19n+m$, where the remainder m is now taken in the range from 1 to 19 (and not from 0 to 18 as is customary), then it is a Hebrew leap year if and only if the following procedure produces a multiple of 3:

- write the remainder m as a two-digit number, e.g., if $y = 1928$ then write $m = 9$ as 09;
- subtract the tens digit from the ones digit, and add 1. For example, (9-0)+1=10 is not a multiple of 3. Thus 1928 was not a Hebrew leap year.

As another example: 1995 is a multiple of 19, so $m = 19$. The procedure produces $(9-1)+1 = 9$, a multiple of 3.

The formula tells us the leap years are precisely those for which $m = 2, 5, 8, 10, 13, 16$, or 19. From this, and the fact that the lengths of Hebrew months generally alternate between 29 and 30 days, Conway and I worked out a fairly fast system to convert to and from Hebrew calendar dates, up to an error of generally at most a day. It did not hurt that we worked this out in the spring of 1995, which is a multiple of 19, and a number of Hebrew months started on the first day of the Roman month around then.

References

[1] G. Kolata. *Scientist at Work: John H. Conway—At home in the elusive world of mathematics*. The New York Times, Science section, October 12, 1993.

[2] S. Roberts. *John Horton Conway: the most charismatic mathematician in the world*. The Guardian, July 23, 2015.
https://www.theguardian.com/science/2015/jul/23/john-horton-conway-the-most-charismatic-mathematician-in-the-world.

[3] S. Roberts. *Genius at Play: The Curious Mind of John Horton Conway*. Bloomsbury, New York, 2015.

[4] J. Conway, G. Agus, and D. Slusky. *Converting between dates in the Hebrew and Roman calendars*. The College Mathematics Journal, 51(6): 466–475, 2020. doi:10.1080/07468342.2020.1814651.