POW #2 – Happy Birthday! Charlotte, Lydia, Vatsal (Queens)

Problem Restatement

Most people know their birthdate – the exact month, day, and year of their birth. However, considerably fewer know what day of the week they were born on. Therefore, how can someone find out what day of the week they were born on from their birthdate alone? (Assume the birthdate is January 1st, 1901, or later).

Process

One attempt we tried was coming up with an equation for conversion of the day of your birthday in 2024 to the day of your birthday in any given year. The equation was b=-y/4-y+d, where y=years since birth, d=day of the week born this year, and b=day of the week you were born on. We came up with this equation by focusing on the days of the week. Since there is a leap year every 4 years, we divided the total number of years since birth by 4. We added this to the number of years since birth to account for the change in day of the week between 2024 and your birthday. We subtracted the value of y/4+y from the day of the week you were born on in 2024 to get the day you were born on in your birthday. This, however, did not produce an accurate result so we moved onto a new method. The issue may be that this method does not account for discrepancies in month length. Also, we assumed that we knew the day of the week a birthday fell on in the current year, which was not given.

For the final solution, we decided to mark the days of the week with numbers 0 through 6 starting at Sunday, to match up with numbers in mod 7.

We knew October 1st was a Tuesday from the given October 2024 calendar.

Next, we needed to convert from October 1st to the 1st of the person's birthday month. To convert from October 1st to the 1st of any month in 2024, we needed to find the number of days that would be added to October 1st, and convert this to a certain number of weekdays to shift forward from October 1st. The way weekdays repeat in a cycle of 7 made us think the next logical step would be to use modular arithmetic.

We started by shifting to November 1st. This would be equivalent to adding 31 days to October 1st. Since 31 is equivalent to 3 in mod 7, we see that this would result in a forward shift of 3 weekdays. Then, for December 1st, we would be adding 30 days to the calculation for November 1st. In mod 7, this is equivalent to 2, we see that this would add an extra 2 weekdays, which we add to the calculation for November 1st to get a total forward shift of 5 weekdays.

For the months prior to October, we used similar logic. To shift to September 1st, we subtracted 30 days. This is equivalent to subtracting 2 weekdays. However, since we are working with modular arithmetic, we can simplify this even further so we are only working with addition, giving us a forward shift of -2 + 7 = 5 weekdays.

The rest of the months were calculated similarly, starting with the shift from the previous month and adding the shift for the next month, working in mod 7 to keep the values manageable. The results can be seen in step 2 of the solution for each month. This was added to the calculation of October 1st to get the total for the 1st of the birthday month in 2024.

After this, it was necessary to have the person add the forward shift in days from the first of their birthday month in 2024 to their actual birthday in 2024. This was accomplished by having the person add one less than the day of the month they were born to the previous answer, as we started with the first of the month.

We next had the person calculate the number of years since they were born with the formula (current year-birthday year) due to the logic surrounding how the weekday of a certain date changes each year; it becomes one day later (i.e. Tuesday to Wednesday) each passing normal year, as the number of days in a year, 365, is equivalent to 1 in mod 7. As a result, this would give the total number the weekday should be shifted back by if we only accounted for normal years.

The calculation of the number of leap years since the person was born ((current year-birthday year)/4) was mainly implemented in the solution to encompass the fact that from the previous normal year to a leap year, a date shifts 2 days rather than 1. Since we only shifted back by 1 for each of these years in the previous step, we need to add another 1 for these special years. Since leap years only occur every 4 years, it is necessary to divide the amount of years between the final year and the birthday by 4 in order to get the value to add to the previous step. During this step, it is also essential to round properly, as half of a leap year doesn't exist. If the user was born in a leap year, this is irrelevant; the value for the number of leap years will be an integer. In leap years, this calculation will be correct no matter when the birthday is in the year. Either the leap day in the birth year will be added or the leap day in 2024 will be added depending on whether the birthday is before February 29th, resulting in one extra leap year in addition to the number of leap years between the two.

If the year wasn't a leap year, the result would be a decimal. Whether we round up or down depends on if the birthday is before or after February 29th (really on or before February 28th or on or after March 1st, as in a non- leap year, February 29th doesn't exist). If it is before, we round down, as the leap day in 2024 is not between the user's birthdate and their birthday in 2024. If it is after, we round up, to account for the leap day in 2024 needing to be included in the calculation.

One error we made at first was rounding down no matter what. This happened because all of our test cases for dates after February 29th were leap years and we didn't realize, so the only value we realized we needed to round for was before February 29th, and we over-generalized our solution of rounding down to all months, giving an inaccurate result for non-leap year birthdates after February 29th .

Since we are shifting back, we then needed to subtract these values from the previous total. Finally, we converted this to mod 7 to figure out the final weekday.

Solution

Sunday = 0, Monday = 1, Tuesday = 2, ... Saturday = 6

1. Start with October 1st, 2024. This is a Tuesday, so we start with the number 2.

2. Find the month of your birthday below and add the corresponding number to your answer from step 1. You may do the calculations for the following steps in normal arithmetic.
   January: +6
   February: +2
   March: +3
   April: +6
   May: +1
   June: +4
   July: +6
   August: +2
   September: +5
   October: +0
   November: +3
   December: +5

3. Add the day of the month you were born on minus 1 to your answer from step 2.

4. Subtract the year you were born from the current year, 2024. This is the number of years since your birth.

5. Take your answer from part 4 and divide it by 4. If you get a non-integer number, round DOWN if the birthday is before February 29th and round UP if the birthday is after February 29th. This answer is the number of leap years since your birth.

6. Add your answers to parts 4 and 5 together. You will use this value in the next step.

7. Start with your answer from part 3. Then, subtract one for every year since your birth and an additional one for every leap year. This means you should take your answer from part 3 and subtract your answer to part 6 from it.

8. Start with your answer from part 7. If you know how to express this in mod 7, do so and skip to step 9. Otherwise, follow these instructions: If the number is positive, take the remainder when it is divided by 7 and go to step 9. If the number is negative, add 7. Repeat the last step until the number you get is positive. Then, take the remainder when this positive number is divided by 7 and go to step 9.

9. Use the following instructions to convert the number between 0 and 6 you got in part 8 to a weekday:
   0 = Sunday
   1 = Monday
   2 = Tuesday
   3 = Wednesday
   4 = Thursday
   5 = Friday
   6 = Saturday

EXAMPLE / TEST CASE 1

Lydia – Tuesday, August 12th, 2008

1. Start with 2.
2. Add 2 to 2 to get 4.
3. (12 – 1) = 11. Add 11 to 4 to get 15.
4. (2024 – 2008) = 16 years.
5. 16/4 = 4 leap years.
6. 16 + 4 = 20.
7. 15 – 20 = -5.
8. Convert to mod 7 by adding 7: -5 = 2 mod 7.
9. 2 = Tuesday. TRUE.

EXAMPLE / TEST CASE 2

Martin Luther King, Jr. – Tuesday, January 15th, 1929

1. Start with 2.
2. Add 6 to 2 to get 8.
3. (15 – 1) = 14. Add 14 to 15 to get 29.
4. (2024 – 1929) = 95 years.
5. floor(95/4) = 23 leap years.
6. 95 + 23 = 118.
7. 29 – 118 = -89.
8. Convert to mod 7 by adding 91: -89 = 2 mod 7.
9. 2 = Tuesday. TRUE.

EXAMPLE / TEST CASE 3

Mrs. Burns – Tuesday, November 25th, 1986

1. Start with 2.
2. Add 3 to 2 to get 5.
3. (25 – 1) = 24. Add 24 to 5 to get 29.
4. (2024 – 1986) = 38 years.
5. ceil(38/4) = 10 leap years.
6. 38 + 10 = 48.
7. 29 – 48 = -19.
8. Convert to mod 7 by adding 21: -19 = 2 mod 7.
9. 2 = Tuesday. TRUE.

Extensions

1. Find all possible years someone could have been born in, given the month and day of their birth and the weekday they were born on.
   - This could involve doing out the calculations of the original solution while leaving the birth year as a variable, and setting the resulting expression equal to all possible normal arithmetic equivalents of the mod 7 representation of the weekday the person was born. Then, we would be able to solve for the birth year(s).

2. Find the weekday someone was born on, given the same information as the original problem, but accounting for the rule that years that are divisible by 100 but not 400 are not leap years.
   - This would have to involve calculating the number of years that are divisible by 100 but not 400 and adding it to the final sum, to cancel out those years' inclusion as leap years.

3. Find the weekday that a date in the future (up to 2099) will occur on, given the month, day, and year.
   - The solution for this extension would involve switching the rounding of the leap year calculation, so the person would round UP if the day is before February 29th and DOWN if the day is after February 29th. This is because the leap day in 2024 would be included only if the starting date is prior to February 29th. In addition, the number of years and leap years would be added to the prior value, not subtracted from it, as the shift is now going forward.

4. Given the month, day, and year of someone's birthday, find out the weekday and the ordinal number of that weekday in the month they were born and in the year they were born (e.g. the first Tuesday, the second Tuesday, and so on).
   - After finding the weekday as usual, it is then possible to figure out which prior days of that calendar month were also on that weekday by repeatedly subtracting 1 week or 7 days from the day until you get a day below 1. Then, it would be easy to figure out where in the order the person's birthday was. Similar steps could be taken to find the weekday of the first of each month and determine how many of each weekday were in each month, then calculating the amount of instances of the particular weekday which occurred prior to the person's birthday.

5. Create a flow chart for the process used to solve each of the problems above.
    a. For the flow chart be sure to use the standard shapes and arrows as listed below:
        i. The start and end blocks should use ovals.
        ii. Arrows should be drawn in between all shapes to show the proper order in which the procedure should be carried out.
        iii. Parallelograms should be used to represent any inputs or outputs that are necessary (i.e., a number being plugged into an equation).
        iv. Rectangles should be used to represent any process (i.e., a mathematical calculation/logical processing).
        v. Diamonds should be used to represent any decisions that must be made (e.g., "Is the birthday before or after July 1st?").
6. Write code that implements the process used to solve each of the problems above.