Solution for Weekly Proof 5

a) To determine the value of D for each month, take the number of days that have occurred before that month and reduce it modulo 7. For example, before July 1st, there have been a total of $31+28+31+30+31+30 = 181$ days (from January 1st to June 30th), and $181 \equiv 6 \pmod 7$. Thus, July corresponds to 6 in the table. We do the same for all the other months to determine the correct value for D.

b) In our algorithm, B corresponds to the number of leap years that have taken place since 1900. For example, if the year is 1930, there have been $\left\lfloor \frac{30}{4} \right\rfloor = 7$ leap years from 1900 to 1930. Specifically, there have been seven February 29th's, which have occurred in the years 1904, 1908, 1912, 1916, 1920, 1924, and 1928, and the formula takes this into consideration. So if the year is a leap year (e.g. 1928), and the month is January or February, the leap day in that year has not yet happened, but the algorithm has mistakenly compensated for that day by adding 1 to the value of B. That is why we need to subtract 1 from the total value of S if we have a leap year and the month is January or February to correct this error.

c) There are several different ways to do this. Here's the El Cheapo way: By our algorithm, December 31st, 1999 was a Friday (or we can just check on any calendar from last year). So January 1st, 2000 must have been a Saturday. If we try January 1st, 2000 on our algorithm, we get Sunday, and so our formula must be pushed back one day in order for it to work correctly for all dates in the 2000's. Thus, we must subtract 1 from S, and that is all we need to do.

Here's a more formal approach. If we changed the definition of A from *the last two digits of the year* to *the year minus 1900*, the trick would work until the year 2100. That is because every fourth year is a leap year, except for the years 1900, 2100, 2200, 2300, 2500, 2600, and so on. So the algorithm would work just fine.

So for the year 2000, we would have $A = 100$ and $B = 25$, rather than $A = 0$ and $B = 0$. In other words we must add 125 to the value of S and then determine the remainder upon division by 7. But $125 \equiv -1 \pmod 7$, and so if we subtract 1 from the value of S before we divide by 7, then the algorithm will produce the correct day of the week.