Problem 2
Year 2000

Input file : y2k.in Output file : y2k.out

OK, you knew there had to be a Y2K problem, so here it is.

You are given a document containing text and numerical data, which may include dates. Your task is to identify (two-digit) years and to reprint the document with these two-digit years replaced by four-digit years. You may assume that any year numbered 24 or less is in the 2000's, while any year numbered 25 or more is in the 1900's (e.g. 16 represents the year 2016 and 26 represents the year 1926*). Yes, we know this rule may imply that your grandmother hasn't been born yet.*

Your program is to recognize dates in any of three formats:

```
dd/mm/yy
yy.mm.dd
Month,dd,yy
```

where *dd* is a two-digit day between 01 and 31, *mm* is a two-digit month between 01 and 12, *yy* is a two-digit year between 00 and 99, and *Month* is one of: January, February, March, April, May, June, July, August, September, October, November, December. The first two formats contain no spaces, and the third format contains a single space after *Month* and a single space after the comma.

Dates should appear on a single line. Dates traversing two lines or dates immediately adjacent to a letter of the alphabet or a digit should not be recognized. Non-existent data such as February 30, 99 do not need to be checked for.

The first line of input to your program will contain a positive integer *n*, indicating the number of lines of text to follow. In each of the lines of text you are to find all dates that occur in any of the formats above, and replace the two-digit year by the appropriate four-digit year as described above.

Sample Input

```
4
Before 02/03/04, but not after December 19, 99,
there was a rehash of the 55.34.02 meeting.  A date, like November 15,
95 cannot traverse two lines, nor can it be surrounded by alphabetics
or numerics like this:  78November 01, 88, or 6801/12/03, or 02/03/04x.
```

Output for Sample Input

```
Before 02/03/2004, but not after December 19, 1999,
there was a rehash of the 55.34.02 meeting.  A date, like November 15,
95 cannot traverse two lines, nor can it be surrounded by alphabetics
or numerics like this:  78November 01, 88, or 6801/12/03, or 02/03/04x.
```

QOJ adaptation notes

This judge reads the number of document lines, followed by those lines, from standard input and writes the converted document to standard output. The third date format is Month dd, yy (one space after the month and after the comma), as explained and demonstrated in the original statement. The original pages are unchanged.

The official marking instructions mention 13 recognized dates, but the supplied official input and printed answer contain 14 conversions. QOJ reconstructs the 13 conversion marks across all 14 actual dates: the first 13 dates are worth 0.928571 marks each and the final date is worth 0.928577, totaling exactly 13. This reconstruction is not the historical weighting. Dates are numbered in document order.

The remaining two marks reward preservation of the rest of the document. Whitespace-separated tokens are aligned against the original and corrected document using minimum insertion, deletion or substitution distance. Either the original or correctly converted form of a date token incurs no preservation error. Zero other token edits earns both preservation marks; one edit earns one mark; two or more edits earns none. A date earns its conversion mark when its aligned token has the corrected value. Whitespace layout is ignored. Total score: 15.