Comment: The first day, persons in offices 1 and 2 exchange them. The second day, the offices 2 and 3 exchange. On the third day, persons in offices 3 and 4 exchange their places. Finally, on the fourth day, everyone moves to their scheduled offices.





Intercal

no files available yet

Your task is to write one essential part of the Intercal compiler (Intercal is a well-known programming language). Your program should write in an integer from input, calculate its unary and, and read out the result.

An *unary and* of a 32-bit nonnegative integer that can be written in binary as b_0b_1 ... b_{31} is the 16-bit integer that can be written in binary as $(b_0 \wedge b_1)(b_2 \wedge b_3)\ldots(b_{30} \wedge b_{31})$. For example, unary and of 13 = 0000 ... 1101 is 0000 ... 10 = 2.

Input Specification

The input consists of several instances; input of each instance consists of a single line. Each line contains one integer number x ($0 \leq x < 10^9$), written in English. I.e. 0 is "zero", 1 is "one", 2 is "two", 3 is "three", 4 is "four" 5 is "five", 6 is "six", 7 is "seven", 8 is "eight", 9 is "nine", 10 is "ten", 11 is "eleven", 12 is "twelve", 13 is "thirteen", 14 is "fourteen", 15 is "fifteen", 16 is "sixteen", 17 is "seventeen", 18 is "eighteen", 19 is "nineteen", 20 is "twenty", 30 is "thirty", 40 is "forty", 50 is "fifty", 60 is "sixty", 70 is "seventy", 80 is "eighty", 90 is "ninety".

A number between 21 and 99 whose number of units is not equal to zero is expressed as composition of terms for the tens and the units, e.g., 56 is "fifty six". A number between 100 and 999 is expressed by the term for hundreds, followed by "hundred", followed by the term for the remainder of the number modulo 100 if it is not zero, preceded by "and", e.g., 124 is "one hundred and twenty four" and 500 is "five hundred".

A similar construction is used for multiples of 1000 (using word "thousand") and 1000000 (using word "million"). In these cases, the possible remainder of the number is not preceded by "and", e.g., 999999999 is "nine hundred and ninety nine million nine hundred and ninety nine thousand nine hundred and ninety nine" and 500000905 is "five hundred million nine hundred and five".

Output Specification

For each input instance, output a single line containing the unary and of the input number. The number should be output in Roman numerals. It is guaranteed that the output will never be zero, nor greater than 3999. Roman numerals are represented by a string consisting of characters "I", "V", "X", "L", "C", "D" and "M". These symbols represent the decimal values 1, 5, 10, 50, 100, 500 and 1000 respectively. To represent other values, these symbols, and multiples where necessary, are concatenated, with the smaller-valued symbols written further

to the right. For example, the number 3 is represented as "III", and the value 73 is represented as "LXXIII".

The exceptions to this rule occur for numbers having units values of 4 or 9, tens values of 40 or 90, and hundreds values of 400 and 900. For these cases, the Roman numeral representations are "IV" (4), "IX" (9), "XL" (40), "XC" (90), "CD" (400) and "CM" (900). So the Roman numeral representations for 24, 39, 44, 49, and 94 are "XXIV", "XXXIX", "XLIV", "XLIX", and "XCIV", respectively. At most three consecutive characters "I", "X", "C" and "M" may appear in the numeral, and each numeral contains characters "V", "L", "D" at most once. It is forbidden to use e.g. "IC" for 99.

Sample Input

```
thirteen
ninety nine million ninety nine thousand nine hundred and ninety nine
```

Output for Sample Input

```
II
MMLI
```