Advanced Problem

Palindrome Numbers

14 March 2009

Source File	`palnum.{java,c,cc}`
Input File	`palnum.in`
Output File	*standard output*

A palindrome is a word, number, or phrase that reads the same forwards as backwards. For example, the name *anna* is a palindrome. Numbers can also be palindromes (e.g. *151* or *753357*). Additionally numbers can of course be ordered by increasing size. The first few palindrome numbers are: *1*, *2*, *3*, *4*, *5*, *6*, *7*, *8*, *9*, *11*, *22*, *33*, ... The number *10* is not a palindrome (even though you could write it as *010*) but a zero as leading digit is not allowed.

But, what is the 17^{th} palindrome?

Input

The input consists of a series of lines with each line containing one integer value i with $1 \leq i \leq 2 * 10^9$. This integer value i indicates the index of the palindrome number that is to be written to the output, where index 1 stands for the first palindrome number (1), index 2 stands for the second palindrome number (2) and so on. The input is terminated by a line containing $i = 0$.

Output

For each line of input (except the last one) exactly one line of output containing a single integer value is to be produced. Given input value, i, the output line is the i^{th} palindrome number.

Output is emitted to standard output, with no leading or trailing spaces.

C, C++	`stdout`
C++	`cout`
Java	`System.out`

Example

Sample input and output are given in figures 1 and 2, respectively.

```
1
12
24
0
```

Figure 1: Input

```
1
33
151
```

Figure 2: Output