Extra Credit – A different kind of Binary, **NEW BINARY**

When a number is expressed in decimal, the *k*-th digit represents a multiple of 10^k. (Digits are numbered from right to left, where the least significant digit is number 0.) For example,

$$81307_{10} = 8\times10^4+1\times10^3+3\times10^2+0\times10^1+7\times10^0 = 80000+1000+300+0+7 = 81307.$$

When a number is expressed in binary, the *k*-th digit represents a multiple of 2^k. For example,

$$10011_2 = 1\times2^4+0\times2^3+0\times2^2+1\times2^1+1\times2^0 = 16+0+0+2+1 = 19.$$

In n**ew binary**, the *k*-th digit represents a multiple of 2^{k+1} - 1. The only possible digits are 0 and 1, except that the least-significant nonzero digit can be a 2. For example,

$$10120_{skew} = 1\times(2^5-1)+0\times(2^4-1)+1\times(2^3-1)+2\times(2^2-1)+0\times(2^1-1) = 31+0+7+6+0 = 44.$$

The first 10 numbers in new binary are 0, 1, 2, 10, 11, 12, 20, 100, 101, and 102. (New binary is useful in some applications because it is possible to add 1 with at most one carry. However, this has nothing to do with the current problem.)

Input

The program should ask for a file to utilize as input. The input file contains one or more lines, each of which contains an integer *n*. If *n* = 0 it signals the end of the input, and otherwise *n* is a nonnegative integer in new binary.

Output

For each number, output the decimal equivalent. The decimal value of *n* will be at most 2^{31} - 1 = 2147483647.

Sample Input

```
10120
20000000000000000000000000000000
10
10000000000000000000000000000000
11
100
11111000001110000101101102000
0
```

Sample Output

```
44
2147483646
3
2147483647
4
7
1041110737
```