Problem 17 - Number Letter Counts

Gautam Manohar

11 June 2018

This document originally appeared as a blog post on my website. Find it at gautammanohar.com/euler/17.

1 Problem Statement

Given a number, write it in English words.

2 My Algorithm

The Hackerank bounds for this problem are $N \leq 10^{12}$. First, we create a dictionary of all the "special" number names that cannot be procedurally generated:

$$0, 1, \ldots, 9, \quad 10, 20, \ldots, 90, \quad 10^2, 10^3, 10^6, 10^9, 10^{12}$$

Now to process the number N. We begin by breaking the number into five blocks of three digits each, starting from the ones digit. These will have, in increasing order, no suffix, the suffix `thousand,''million," `billion,''and`trillion." We then find the name of the number corresponding to each of these blocks and add the corresponding suffix.

We maintain an array of strings which we shall join together to make the name. We will use two tricks. First, that the integer division of N by 10^k removes the last k digits of N; and second, that N modulo 10^k returns the last k digits of N.

Now we define a function that returns the name of a number with at most three digits. If N is zero, we return an empty string. If N is in our look-up table, we are done. If not, we get the first digit, or hundreds digit. If it is non-zero, we add its name and "hundred" to the array. Then we get the last two digits. If the corresponding number is in the look-up table, we add it to the array. If not, we find the tens digit and if it is non-zero, we add the name of the corresponding multiple of ten (from our look-up table) to our array. Finally, we add the name of the ones digit if it is non-zero.

Then, we join together the three-digit name with suffix for each block of three digits in the number. This solution has time complexity $O(\log n)$. Because we process the number in blocks of three, the constant factor is small.