**Technology details:**

**Asp .net MVC 4.0 in Visual Studio 2017 RC for MAC**

**Explanation**

My program will read the input. Input in this program will be treat as string type variable. From this string, it will be divided to array. In those arrays, each arrays will contain maximum 3 chars. The process for dividing string is dividing arrays from behind. If I have string with value “1234”, I will have two arrays 1 and 234. Form each array, the pattern of number in English Spell will read by system. The patterns in those number are single digit, tens digit, hundreds digit, Thousands and zero contains. Single number is number with original pronunciations such as 1 = one, 3 = three, etc. Tens digit is number in ten multiplication such as 20=twenty, 30 = thirty, etc. Hundreds digit is number in first array except zero, such as 100 = one **hundred**, 304= three **hundred** four, etc. Thousands is condition using words thousand, million, billion, etc. I used that word because this word is multiplication of thousand. It depend on number of character in input string. If I have 3 character in each array, and thousands word used every three character in input, it condition will be checked depends on number of array. If I have 3 arrays I will have thousand and million. If I have 4 array, I will have thousand, million, and billion. Last one is zero contains. Special zero contains number like 10 (ten), 1000, etc. This approach is very simple, but some case result improper solution. Improper result such as the use of word AND. For the cents digit, only receive 2 digit input.

This method is simple. Easy to implement, but seen brute force solution. But this method created from observation of pattern on series of number. In other solution, we can read one by one char character and translate it based on number of character and where it index (place). Of course we still have rule for special number like 10,11,12,13, tens digit, “hundreds” condition, “thousands” condition and leading zero to avoid miss read the input. If we compare the two methods, both look similar. Because, both used pattern observation. If we have fixed pattern, my solution will easy to use and implementation, also have minimum possibility to have error.

There are limitations for my implementation. Input no more 21 digits char on dollar and repeat. All input must be number character. Format input will be xxx.yy, x for number will be translate into number on dollar, after that separated by “.”, after that yy for number on cents. Numbers in cents has maximum 2 numbers.

Maximum number input : 999999999999.99

Results : Nine Hundred Ninety-Nine Billion Nine Hundred Ninety-Nine Million Nine Hundred Ninety-Nine Thousand Nine Hundred Ninety-Nine Nine Hundred Ninety-Nine Nine Dollars And Ninety-Nine Cents

I use reference on <http://en.wikipedia.org/wiki/Names_of_large_numbers>, If you want to increase upper bound, replace it with the number on power number in column short scale.