Hackerrank Codesprint 5

Problem 1: isFibo?

For a given number, can you find if the number is Fibonacci number?

Constraint were given about the number to be lesser than 10000000000(10 zeroes).

Approach 1: The most general one, the fact that a number N is a Fibonacci if and only if either 5\*N\*N+4 or 5\*N\*N-4 is a perfect square. This is most elegant way to proceed for the answer and the fact that the square of the number lying in the given constraints lies in the range covered by *long long int* of C/C++ does make this a preferred choice.

Approach 2: Another fact that Fibonacci number increase exponentially, it leaves us with only around 50 fibonacci numbers in the given range. So another approach to take would be to go for computing all these numbers and then perform the binary search.

CODE: <https://github.com/gangak2/Hackerrank/blob/master/Codesprint%205/isFibo.cpp>

Problem 2: matrixTracing

Given below is a word from the English dictionary arranged as a matrix

MATHE  
ATHEM  
THEMA  
HEMAT  
EMATI  
MATIC  
ATICS

Tracing the matrix is starting from the top left position and at each step move either RIGHT or DOWN, to reach the bottom right of the matrix. It is assured that any such tracing generates the same word. How many such tracings can be possible for a given word of length m+n-1 written as a matrix of size m \* n?

This is essentially finding the number of paths from the top left corner to bottom right corner. For a given mXn grid, there would be m-1 RIGHT moves + n-1 DOWN moves. Another fact is that any permutations of these moves will take us to the bottom left corner. ☺

So the our approach simply involves placing m-1 RIGHT moves on m+n-2 places, (The rest of the places are occupied by the DOWN moves), i.e. {m+n-2} choose {m-1} (={m+n-2} choose {n-1}).

Answer can be large, so answer%1000000007 was supposed to be reported, so necessary precautions in computing the binomial were supposed to be taken.

CODE: <https://github.com/gangak2/Hackerrank/blob/master/Codesprint%205/matrixTracing.cpp>

Problem 3: spcialMultiple

You are given an integer N. Can you find the least positive integer X made up of only 9’s and 0’s such that X is a multiple of N?

Approach 1: Since X consists of 9 and 0, it is divisible by 9. Now if the given N is not divisible by 9, multiply it with 9. Now keep on adding this number to itself till you reach the number consisting entirely of 0s and 9s.

This is almost as a brute force method and takes too much time and also the possibility of overflow can’t be overlooked☹

Approach 2: Recursive backtracking. ☺

CODE: <https://github.com/gangak2/Hackerrank/blob/master/Codesprint%205/specialMultiple.cpp>