

The screenshot shows the LeetCode problem page for "202. Happy Number". The top navigation bar includes links for Description, Note, Editorial, Solutions, and Submissions, along with various user and search icons. Below the title, there are three tabs: Easy (selected), Topics, and Companies. The main text asks to write an algorithm to determine if a number n is happy. It defines a happy number as one that eventually reaches 1 through the process of summing the squares of its digits. It also notes that numbers that loop endlessly in a cycle are not happy. The example provided for $n = 19$ shows the sequence: $1^2 + 9^2 = 82$, $8^2 + 2^2 = 68$, $6^2 + 8^2 = 100$, and $1^2 + 0^2 + 0^2 = 1$. The output for $n = 2$ is false, as it enters a cycle at 4. The constraints section indicates that n is a 32-bit signed integer ranging from -2³¹ to 2³¹-1. The bottom of the page shows social sharing icons for 11.7K likes, 391 comments, and other options.

202. Happy Number

Easy Topics Companies

Write an algorithm to determine if a number n is happy.

A **happy number** is a number defined by the following process:

- Starting with any positive integer, replace the number by the sum of the squares of its digits.
- Repeat the process until the number equals 1 (where it will stay), or it **loops endlessly in a cycle** which does not include 1.
- Those numbers for which this process **ends in 1** are happy.

Return `true` if n is a happy number, and `false` if not.

Example 1:

Input: $n = 19$
Output: true
Explanation:
 $1^2 + 9^2 = 82$
 $8^2 + 2^2 = 68$
 $6^2 + 8^2 = 100$
 $1^2 + 0^2 + 0^2 = 1$

Example 2:

Input: $n = 2$
Output: false

Constraints:
-2³¹ ≤ n ≤ 2³¹-1

11.7K likes 391 comments