

202. Happy Number

Add to List ▾

Question

Editorial Solution

My Submissions (/problems/happy-number/submissions/)

Total Accepted: **105678** Total Submissions: **268662** Difficulty: **Easy** Contributors: **Admin**

Write an algorithm to determine if a number 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, and 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 numbers.

Example: 19 is a happy number

$$1^2 + 9^2 = 82$$

$$8^2 + 2^2 = 68$$

$$6^2 + 8^2 = 100$$

$$1^2 + 0^2 + 0^2 = 1$$

Credits:

Special thanks to @mithmatt (<https://leetcode.com/discuss/user/mithmatt>) and @ts (<https://leetcode.com/discuss/user/ts>) for adding this problem and creating all test cases.

Subscribe (/subscribe/) to see which companies asked this question

Show Tags

Show Similar Problems

Have you met this question in a real interview? ☐ Yes ☐ NoDiscuss (<https://discuss.leetcode.com/category/210>)

Top Solutions

Pick One (/problems/random-one-question/)

Notes

Python ▾



```
1 class Solution(object):
2     def isHappy(self, n):
3         """
4         :type n: int
5         :rtype: bool
6         """
7
8         while n > 6:
9             sum = 0
10            while n:
11                sum = sum + (n%10)**2
12                n = n//10
13            n = sum
14        return n == 1
15
```

Custom Testcase ☐

Contribute Testcase

Run Code

Submit Solution

[Frequently Asked Questions \(/faq/\)](#) | [Terms of Service \(/tos/\)](#)[Privacy](#)

Copyright © 2017 LeetCode