Question: https://leetcode.com/problems/happy-number/

So what we do is we maintain a hashedSet and perform the iterative sum of square operation until we get 1 or we encounter a number which we have already encountered before.

In case we try to add an existing element to hashedSet it will return false.

Code:  
class Solution {

public boolean isHappy(int n) {

Set<Integer> mem = new LinkedHashSet<Integer>();

while(n!=1 && mem.add(n)){

n = sumOfSquares(n);

}

if(n==1)return true;

else return false;

}

public int sumOfSquares(int n){

int sum = 0;

while(n>0){

int x = n%10;

sum+=x\*x;

n/=10;

}

return sum;

}

}

Github Link :<https://lnkd.in/ecwtJeaz>