

# **Modified Kaprekar Numbers**



opics	Topics	Editorial	Discussions	Leaderboard	Submissions	Problem	
opics	Topics	Editorial		Leaderboard		Problem	

A modified *Kaprekar number* is a positive whole number n with d digits, such that when we split its square into two pieces – a right hand piece r with d digits and a left hand piece l that contains the remaining d or d-1 digits, the sum of the pieces is equal to the original number (i.e. l + r = n).

Note: r may have leading zeros.

Here's an explanation from Wikipedia about the **ORIGINAL** Kaprekar Number (spot the difference!): In mathematics, a Kaprekar number for a given base is a non-negative integer, the representation of whose square in that base can be split into two parts that add up to the original number again. For instance, 45 is a Kaprekar number, because  $45^2 = 2025$  and 20+25 = 45.

#### The Task

You are given the two positive integers p and q, where p is lower than q. Write a program to determine how many Kaprekar numbers are there in the range between p and q (both inclusive) and display them all.

#### **Input Format**

There will be two lines of input: p, lowest value q, highest value

#### Constraints

0

# **Output Format**

Output each Kaprekar number in the given range, space-separated on a single line. If no Kaprekar numbers exist in the given range, print INVALID RANGE.

## **Sample Input**

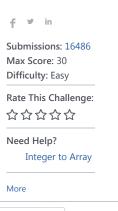
100

### **Sample Output**

1 9 45 55 99

# **Explanation**

1, 9, 45, 55, and 99 are the Kaprekar Numbers in the given range.



```
Current Buffer (saved locally, editable) & 5
                                                                                                                  V 57
 1 ▼ #include <cmath>
     #include <cstdio>
     #include <vector>
     #include <iostream>
 5
     #include <algorithm>
 6
    using namespace std;
 8
    int main() {
 9
10
         /* Enter your code here. Read input from STDIN. Print output to STDOUT */
         int p,q;
11
12
         scanf("%d %d", &p, &q);
13
         \verb|int k[| = { 1, 9, 45, 55, 99, 297, 703, 999, 2223, 2728, 4950, 5050, 7272, 7777, 9999, 17344, }|
14
     22222, 77778, 82656, 95121, 99999
15
16
17
         int n = sizeof(k)/sizeof(int);
         bool hay = false;
18
19
         for (int i = 0; i < n; i++) {
   if (k[i] >= p && k[i] <= q) {
      //console.write(k[i] + " ");</pre>
20 ▼
21
22
                   cout << k[i] << " ";
23
24
                   hay = true;
25
26
27
         if (!hay) {
              cout << ("INVALID RANGE") << endl;</pre>
28
29
30
31
         return 0;
32
33
                                                                                                                   Line: 14 Col: 126
                        Test against custom input
                                                                                                         Run Code
                                                                                                                      Submit Code
1 Upload Code as File
                                          Congrats, you solved this challenge!
                                                          ✓ Test Case #1
                                                                                                   ✓ Test Case #2
                ✓ Test Case #0
                ✓ Test Case #3
                                                          ✓ Test Case #4
                                                                                                   ✓ Test Case #5
                ✓ Test Case #6
                                                                                                              Next Challenge
                                                   Copyright © 2016 HackerRank. All Rights Reserved
```

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature

