BACK   Comfortable Numbers   ⌄

DESCRIPTION   SOLUTIONS 4214   COMMENTS 40   ❯   CODEWRITING   SCORE: 300/300

Let's say that number `a` feels *comfortable* with number `b` if `a ≠ b` and `b` lies in the segment `[a - s(a), a + s(a)]`, where `s(x)` is the sum of `x`'s digits.

How many pairs `(a, b)` are there, such that `a < b`, both `a` and `b` lie on the segment `[l, r]`, and each number feels *comfortable* with the other?

Example

For `l = 10` and `r = 12`, the output should be
`comfortableNumbers(l, r) = 2`.

Here are all values of `s(x)` to consider:

- `s(10) = 1`, so `10` is *comfortable* with `9` and `11`;
- `s(11) = 2`, so `11` is *comfortable* with `9`, `10`, `12` and `13`;
- `s(12) = 3`, so `12` is *comfortable* with `9`, `10`, `11`, `13`, `14` and `15`.

Thus, there are `2` pairs of numbers *comfortable* with each other within the segment `[10; 12]`: `(10, 11)` and `(11, 12)`.

Input/Output

- **[execution time limit] 4 seconds (js)**

- **[input] integer l**

   Guaranteed constraints:
   `1 ≤ l ≤ r ≤ 1000`.

- **[input] integer r**

   Guaranteed constraints:
   `1 ≤ l ≤ r ≤ 1000`.

- **[output] integer**

   The number of pairs satisfying all the above conditions.

[JavaScript (ES6)] Syntax Tips

```
// Prints help message to the console
```

👍 👎 🚩