

PDP-27 (2015) - Camp seniors - 1 (happy)

Time limit: 1.0s Memory limit: 64M

HAPPY

Let's say we have a natural number N . We start from this number and apply the following process: we sum the squares of all its digits, replace the number with the result, and repeat the process. If at some point the result becomes 1 (and stays there), then we say that the number N is **happy**. Conversely, if the process repeats indefinitely without ever resulting in the number 1, then we say that the number N is **sad**.

For example, the number 7 is happy because the process described above leads to the following steps: 7, 49, 97, 130, 10, 1, 1, 1... On the other hand, the number 42 is sad because the process leads to an infinite sequence 42, 20, 4, 16, 37, 58, 89, 145, 42, 20, 4, 16, 37, and so on... We want to be able to answer questions of the form: how many happy numbers exist in the range between numbers A and B (inclusive)?

Input

The first line of input will contain a single natural number K , the number of questions to be asked. Each of the next K lines of input will contain two natural numbers A and B separated by a single space: the boundaries of the range corresponding to a question.

Output

The output should consist of K lines, each containing a single natural number: the number of happy numbers that are the answer to the respective question.

Constraints

$$1 \leq K \leq 1.000$$

$$0 \leq A \leq B \leq 1.000.000.000$$

Execution Time Limit: 1 sec.

Memory limit: 64 MB.

Example of Input - Output

STDIN (happy.in)

```
7
8 10
34 43
13 28
17 17
27 41
0 42
1 1
```

STDOUT (happy.out)

```
1
0
4
0
3
9
1
```

Explanation of the Example: The happy numbers up to 50 are 1, 7, 10, 13, 19, 23, 28, 31, 32, 44, and 49.