Samet Cagri Aktepe – 26.10.2022

https://www.spoj.com/problems/ADDREV/

DDREV - Adding Reversed Numbers

#simple-math #ad-hoc-1

The Antique Comedians of Malidinesia prefer comedies to tragedies. Unfortunately, most of the ancient plays are tragedies. Therefore the dramatic advisor of ACM has decided to transfigure some tragedies into comedies. Obviously, this work is very hard because the basic sense of the play must be kept intact, although all the things change to their opposites. For example the numbers: if any number appears in the tragedy, it must be converted to its reversed form before being accepted into the comedy play.

Reversed number is a number written in Arabic numerals but the order of digits is reversed. The first digit becomes last and vice versa. For example, if the main hero had 1245 strawberries in the tragedy, he has 5421 of them now. Note that all the leading zeros are omitted. That means if the number ends with a zero, the zero is lost by reversing (e.g. 1200 gives 21). Also note that the reversed number never has any trailing zeros.

ACM needs to calculate with reversed numbers. Your task is to add two reversed numbers and output their reversed sum. Of course, the result is not unique because any particular number is a reversed form of several numbers (e.g. 21 could be 12, 120 or 1200 before reversing). Thus we must assume that no zeros were lost by reversing (e.g. assume that the original number was 12).

Input

The input consists of *N* cases (equal to about 10000). The first line of the input contains only positive integer *N*. Then follow the cases. Each case consists of exactly one line with two positive integers separated by space. These are the reversed numbers you are to add.

Output

For each case, print exactly one line containing only one integer - the reversed sum of two reversed numbers. Omit any leading zeros in the output.

Example

```
Sample input:

3

24 1

4358 754

305 794
```

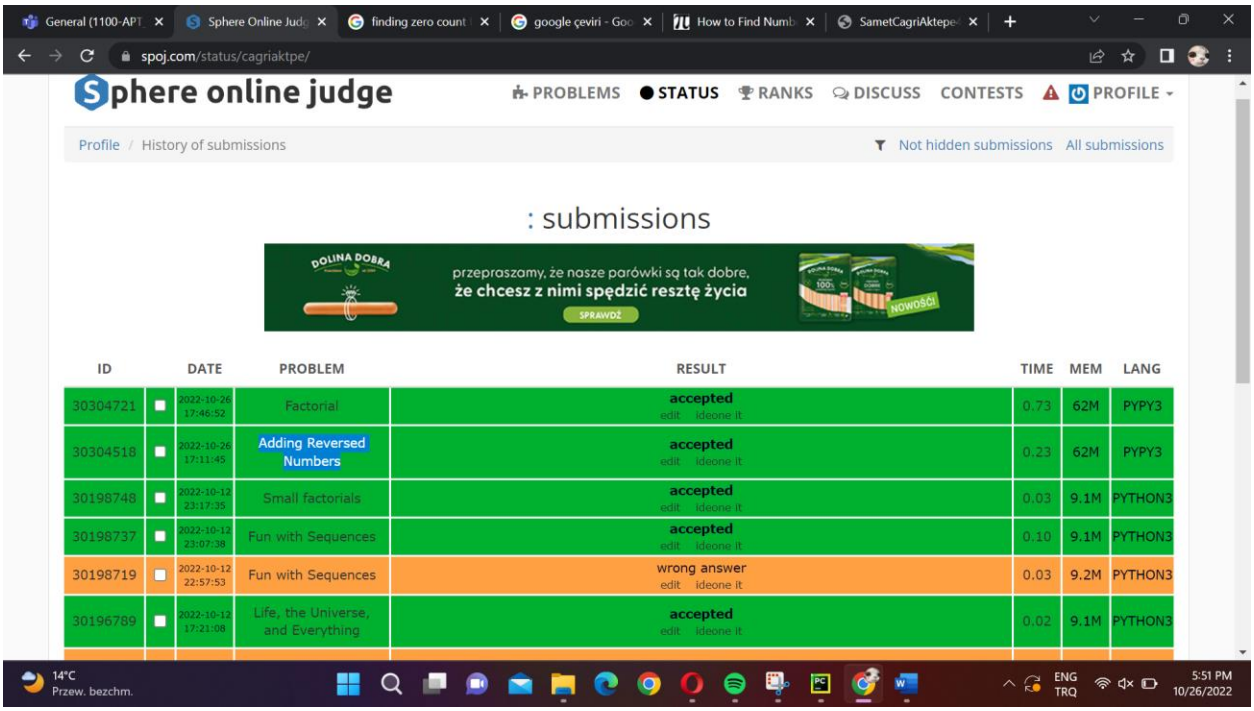
Sample output:

34

1998

1

```
def reverseNumber(number):
    reversed_num = 0
    while number != 0:
        # taking modulo with 10 gives us the last digit of num
        curr_digit = number % 10

        # appending the last digit of num to reversed_num
        # for this we multiply the curr reversed_num by 10 and add curr_digit
to it

        reversed_num = 10 * reversed_num
        reversed_num = reversed_num + curr_digit

        # remove the last digit from num by dividing it by 10
        number = number // 10
    return reversed_num

def addReversedNumbers(input):
    # creating array based on space
    array_input = input.split(" ")
    first_number = int(array_input[0])
    second_number = int(array_input[1])

    # reversing numbers individually
    reversed_first_number = reverseNumber(first_number)
    reversed_second_number = reverseNumber(second_number)

    # summing reversed numbers
    total = reversed_first_number + reversed_second_number

    # reversing total
    reversed_total = reverseNumber(total)

    return reversed_total

# how many input we need
time = int(input())

# for that input count asking input
for i in range(time):
    numbers = input()
    print(addReversedNumbers(numbers))
```


spoj.com/status/cagriaktepe/
Sphere online judge
PROBLEMS STATUS RANKS DISCUSS CONTESTS PROFILE
Profile / History of submissions
Not hidden submissions All submissions
: submissions
przepraszamy, że nasze parówki są tak dobre, że chcesz z nimi spędzić resztę życia
SPRAWDŹ
NOWOŚĆ!
ID DATE PROBLEM RESULT TIME MEM LANG
30304721 2022-10-26 17:46:52 Factorial accepted edit ideone it 0.73 62M PYPY3
30304518 2022-10-26 17:11:45 Adding Reversed Numbers accepted edit ideone it 0.23 62M PYPY3
30198748 2022-10-12 23:17:35 Small factorials accepted edit ideone it 0.03 9.1M PYTHON3
30198737 2022-10-12 23:07:38 Fun with Sequences accepted edit ideone it 0.10 9.1M PYTHON3
30198719 2022-10-12 22:57:53 Fun with Sequences wrong answer edit ideone it 0.03 9.2M PYTHON3
30196789 2022-10-12 17:21:08 Life, the Universe, and Everything accepted edit ideone it 0.02 9.1M PYTHON3
14°C Przew. bezchm.
ENG TRQ
5:51 PM 10/26/2022