Yet Another Electronic Device!!!

Fascinated as he is by the uncanny world of electronics, our friend MKS now decides to launch his own creation → A N-Digit Carry Finder (an analogue of a N-Bit Binary Adder) which can be used to find the number of times we can have a non-zero carry while adding two numbers ($A = A_n A_{n-1} \ldots A_2 A_1$ and $B = B_n B_{n-1} \ldots B_2 B_1$) having **exactly N** digits.

It consists of 'N' Full Decimal Adders. The i-th Full Adder takes as input three digits A_i, B_i and C_{i-1} and outputs a digit C_i(0 or 1), which is the carry generated on adding the digits A_i and B_i and C_{i-1}. ($C_i = 1$ if $A_i + B_i + C_{i-1} > 9$, otherwise 0).

This C_i is now provided to the next (i+1-th) Full Adder in order to be added with the digits A_{i+1} and B_{i+1} and also to the accumulator which as the name suggests accumulates the sum of all C_j ($1 <= j <= i$).

Note: $C_0 = 0$ always and $0 <= A_i, B_i <= 9$.

For Example: Adding two numbers, A = 4567 and B = 734 (or B = 0734), the addition proceeds as shown and the accumulator gets a final value of 3.

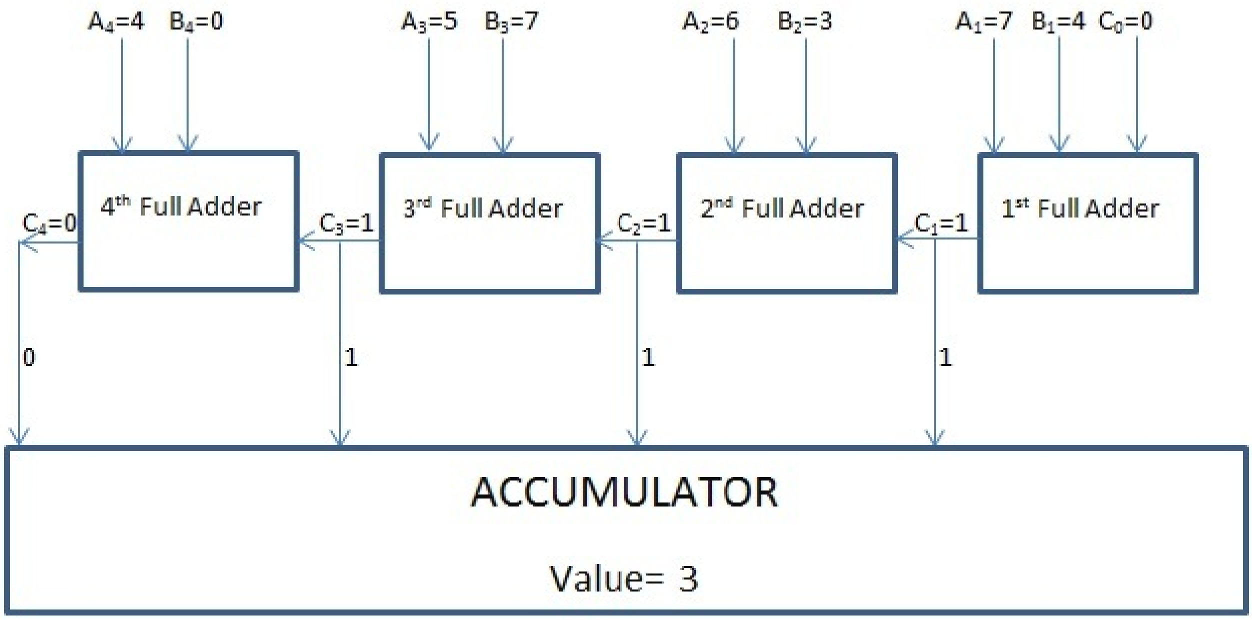
In the 1st Adder, $A_1 = 7$, $B_1 = 4$, $C_0 = 0$ and $A_1 + B_1 + C_0 = 11$. Therefore Carry $C_1 = 1$.

In the 2nd Adder, $A_2 = 6$, $B_2 = 3$, $C_1 = 1$ and $A_2 + B_2 + C_1 = 10$. Therefore Carry $C_2 = 1$.

In the 3rd Adder, $A_3 = 5$, $B_3 = 7$, $C_2 = 1$ and $A_3 + B_3 + C_2 = 13$. Therefore Carry $C_3 = 1$.

In the 4th Adder, $A_4 = 7$, $B_4 = 0$, $C_3 = 1$ and $A_4 + B_4 + C_3 = 8$. Therefore Carry $C_4 = 0$.

The Value in the Accumulator = $C_1 + C_2 + C_3 + C_4 = 3$.



Your task is to find the number of ways of getting a value **K** in the **accumulator** while adding two numbers containing **at most N** digits each. Note that we are adding the numbers in their **base 10** representation. Since the total number of ways can be very large, print your answer modulo

1000000007 (10^9 +7).

Input

The first line of input contains an integer T. Then T lines follow containing two space separated integers N and K.

Output

Print the required answer modulo 1000000007(10^9 +7) in the i^{th} line corresponding to the i^{th} Test case .

Constraints

1 <= T <= 500

1 <= N <= 1000

1 <= K <= N

Example

Input:
```
4
1 1
2 1
2 2
3 3
```

Output:
```
45
4500
2475
136125
```

Explanation

For test case 1, the carry appears when adding:

- 1 and 9, 2 and 9, 3 and 9 ... 9 and 9 = 9 cases.
- 2 and 8, 3 and 8, 4 and 8 ... 9 and 8 = 8 cases.
- 3 and 7, 4 and 7, 5 and 7 ... 9 and 7 = 7 cases.
- ...
- 9 and 1 = 1 case.

There are (9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 + 0) = 45 cases in total and in each case, the carry appears exactly once.