Dortmund Dilemma

by Bidhan

Borussia Dortmund are a famous football (soccer) club from Germany. Apart from their fast-paced style of playing, the thing that makes them unique is the hard to pronounce names of their players (błaszczykowski , papastathopoulos , großkreutz etc.).

The team's coach is your friend. He is in a dilemma as he can't decide how to make it easier to call the players by name, during practice sessions. So, you advise him to assign easy names to his players. A name is easy to him if

1. It consists of only one word.
2. It consists of only lowercase english letters.
3. Its length is **exactly** N.
4. It contains **exactly** K different letters from the 26 letters of English alphabet.
5. At least one of its **proper** prefixes matches with its **proper** suffix of same length.

Given, N and K you have to tell him the number of easy names he can choose from modulo $(10^9 + 9)$.

Note : A prefix P of a name W is proper if, $P \neq W$. Similarly, a suffix S of a name W is proper if, $S \neq W$.

Input Format
The first line of the input will contain T (the number of testcases). Each of the next T lines will contain 2 space separated integers N and K.

Output Format
For each testcase, output the number of ways the coach can assign names to his players modulo $(10^9 + 9)$.

Constraints
$1 \leq T \leq 10^5$
$1 \leq N \leq 10^5$
$1 \leq K \leq 26$

Sample Input #1

```
3
1 1
2 1
4 2
```

Sample Output #1

```
0
26
2600
```

Sample Input #2

```
5
2 2
5 1
3 2
```

```
6 2
1 3
```

Sample Output #2

```
0
26
650
13650
0
```

Submissions:437

Max Score:150

Difficulty: Advanced

Rate This Challenge:

More

Current Buffer (saved locally, editable) Java 7

```
1  import java.io.*;
2  import java.util.*;
3  import java.text.*;
4  import java.math.*;
5  import java.util.regex.*;
6
7  public class Solution {
8
9      public static void main(String[] args) {
10         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11     }
12 }
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code