

Mirko has written the following function:

```
int fun() {
  int ret = 0;
  for (int a = X1; a <= Y1; ++a)
    for (int b = X2; b <= Y2; ++b)
      ...
      for (int <N-th> = XN; <N-th> <= YN; ++<N-th>)
        ret = (ret + 1) % 1000000007;
  return ret;
}

function fun: longint;
var
  ret: longint;
  a, b, ... , y, z: longint;
begin
  ret := 0;
  for a := X1 to Y1 do
    for b := X2 to Y2 do
      ...
      for <N-th> := XN to YN do
        ret := (ret + 1) mod 1000000007;
  fun := ret;
end;
```

$\langle N\text{-th} \rangle$  denotes the  $N^{\text{th}}$  lowercase letter of the English alphabet. Each  $X_i$  and  $Y_i$  denotes either a positive integer less than or equal to 100 000 or a name of a variable that some outer loop iterates over. For example,  $X_3$  can be either a, b, or an integer literal. At least one of  $X_i$  and  $Y_i$  will be an integer literal (i.e. not a variable name) for every  $i$ .

Compute the return value of the function.

### INPUT

The first line of input contains the positive integer  $N$  ( $1 \leq N \leq 26$ ).

For the next  $N$  lines, the  $i^{\text{th}}$  line contains  $X_i$  and  $Y_i$ , separated with a space. If  $X_i$  and  $Y_i$  are both integer literals, then  $X_i \leq Y_i$ .

### OUTPUT

The first and only line of output must contain the return value of the function.

### SAMPLE TESTS

input	input	input
2	3	3
1 2	2 3	1 2
a 3	1 2	a 3
	1 a	1 b
output	output	output
5	10	11