2631 - Unlock My Safe

Description

I forgot the password to my safe. There is a lot of money in it! Please help me unlock the safe. The keypad looks like this.

I do not remember how long my password is. Hence, you need to try a different length of the password. However, there are some hints that I can recall.

•I never use characters *, #, 0 and 9 in my password.
•Each digit in the password is distinct. That is, they never appear more than once.
•My password is at most 8 digits (1 <= N <= 8, where N is a number of digits in the password).
•Each digit i in the password always has the value less than or equal to N (that is, a password 132 is valid for N = 3 but a password such as 124 is invalid because the 3rd digit exceeds 3).

Use the information above and generate all possible permutations. One permutation corresponds to one guess of a password to unlock my safe. Importantly, the correct password is deliberately fixed at position L\3 in the sorted array of permutations, where L is a number of all possible permutations and '\' is an integer division. The sorted array of permutations is in ascending order and the starting index in the sorted array begins at 0 (not 1).

Write a program to find a correct password for a given length (a number of digits in the password).

Input specification

The first line of the input contains an integer T (1 <= T <= 6) denoting the number of test cases. After that T test cases follow. Each test case contains an integer N (1 <= N <= 8) denoting a number of digits in a password.

Output specification

Your program should output the N-digit password for each corresponding test case, one password per line.

Sample input

```
3
2
3
1
```

Sample output

```
12
213
1
```

Hint(s)

There are 3 test cases above. In the second case, for example, the sorted permutations are {123, 132, 213, 231, 312, 321}. Password is located at the position 6\3 = 2 (integer division). When the starting index begins at 0, the password is, therefore, 213.

Source	ACM-ICPC Thailand Southern Programming Contest 2013
Added by	**ymondelo20**
Addition date	2013-11-19
Time limit (ms)	3000
Test limit (ms)	2500
Memory limit (kb)	256000
Output limit (mb)	64
Size limit (bytes)	15000
Enabled languages	Bash C C# C++ Java Pascal Perl PHP Python Ruby Text