

6351 ATM PIN Theft

Gabby the Good Guy is entering his PIN (password) on an ATM while Billy the Bad Buddy is looking over his shoulder to find his PIN. The keys on the ATM keypad are such that when pressed and held, several of that key may actually be entered. The numbers entered are displayed as '*' on the screen, so Billy may not be able to find out the exact PIN. He just knows the keys that Gabby has pressed (in order), but does not know how many times each has been entered into the ATM.

PINs are strings of exactly four digits (0 to 9). ATM has 10 keys for digits, plus one key for Backspace which deletes the last entered digit. Like other keys, if we press and hold Backspace, more than one digit may be deleted. Note that Billy sees in the correct order each key that Gabby has pressed (including the backspace key), but he does not know the number of times that the specified key has actually been entered. Although Gabby's PIN is a string of exactly four digits, he may press more than 4 keys including the keys deleted by Backspace as the ATM accepts any number of keys. He may also press less than 4 keys to produce his 4-digit key.

For example, suppose Billy sees that Gabby has first pressed '1', then '3', then '5', and finally '7'. Since PINs have four digits and no backspace key has been pressed, then there can be only one PIN which is 1357. But in case the pressed keys are '1', '3', and '5' (in that order), the actual PIN may have been 1135, 1335, or 1355.

Input

There are multiple test cases in the input. The first line of each test case has only a single integer n ($0 < n < 10$), which is the number of keys Gabby has pressed. The next line contains n space-separated integers that are either a single digit (0 to 9), or 99 which denotes the Backspace key. These are the sequence of keys that Gabby has pressed. You are guaranteed that the key 99 appears at most once in each test case but other keys may appear more than once.

The input terminates with a line containing '0'.

Output

For each test case, write a single line containing the number of possible PINs consistent with the given sequence of pressed keys.

Of course, if there is no PIN consistent with the given sequence of pressed keys, the output must be '0'.

Sample Input

```
4
1 3 5 7
3
4 6 8
8
1 2 3 4 5 99 3 5
3
2 2 5
0
```

Sample Output

1
3
7
2