DWITE '09 R2 #3 - Binary Test String

Time limit: 2.0s **Memory limit:** 64M

DWITE Online Computer Programming Contest, November 2009, Problem 3

Besides high school programming contests, another application for automated execution of code is in *test suites*, that validate the correctness of software (which, in some ways, is what DWITE judge does anyway). Given enough tests and edge cases in sample input, we could increase the confidence that the software is correct. Absolute certainty will require mathematical proofs, or generating *every possible sample input* and testing against that (which is often not possible).

For this problem, we'll be generating binary strings – words of length 4, consisting of only `1`s and `0`s, with certain additional restrictions specified in the input file.

The input file will contain 5 lines, a binary string of length 1 to 4 – a pattern that should not appear in binary strings in the generated set.

That is: if the input is `1`, then the only valid output string is `0000` (any other binary string of size 4 will contain `1`). Substring `11` disallows: `1100`, `0110`, `0011`; but it also appears in `1110`, `0111`, and `1111`.

The output will contain 5 sets of generated binary strings, sorted in *ascending* order, that is, from `0000` to `1111`.

Sample Input

```
1
11
```

Sample Output

```
0000
0000
0001
0010
0100
0101
1000
1001
1010
```

Problem Resource: DWITE