

Description

Your friend tripped while carrying a tray of computer files through the lab. All of the files fell to the ground and broke. Your friend picked up all the file fragments and called you to ask for help putting them back together again.

Fortunately, all of the files on the tray were identical, all of them broke into exactly two fragments, and all of the file fragments were found. Unfortunately, the files didn't all break in the same place, and the fragments were completely mixed up by their fall to the floor.

The original binary fragments have been translated into strings of ASCII 1's and 0's. Your job is to write a program that determines the bit pattern the files contained.

Input

The input begins with a single positive integer on its own line indicating the number of test cases, followed by a blank line. There will also be a blank line between each two consecutive cases.

Each case will consist of a sequence of "file fragments," one per line, terminated by the end-of-file marker or a blank line. Each fragment consists of a string of ASCII 1's and 0's.

Output

For each test case, the output is a single line of ASCII 1's and 0's giving the bit pattern of the original files. If there are $2N$ fragments in the input, it should be possible to concatenate these fragments together in pairs to make N copies of the output string. If there is no unique solution, any of the possible solutions may be output.

Your friend is certain that there were no more than 144 files on the tray, and that the files were all less than 256 bytes in size.

The output from two consecutive test cases will be separated by a blank line.

Sample Input

```
1

011
0111
01110
111
0111
10111
```

Sample Output

```
01110111
```

Submitting Your Code and Report

Your file name must be "assign1.cpp", and it is the only file you should submit. Please do not submit the executable file of your program. Turn in your project using the "oops submit" command as follows:

```
$ oops submit assign1 assign1.cpp
```

Late submissions will lose 20 points per day.

If you do not follow this submission guideline, you will lose 10 points out of 100.

You should also submit a hard copy of your code to TA. Your report must have a cover page with your student ID and name. In the report, your code must be well commented to explain your algorithm. Also, the sample input and the output of your program must be included in the report.