

Character Recognition

This problem requires you to write a program that performs character recognition.

Details:

Each ideal character image has 20 lines of 20 digits. Each digit is a '0' or a '1'. See Figure 1a for the layout of character images in the file.

The file FONT.DAT contains representations of 27 ideal character images in this order:

□abcdefghijklmnopqrstuvwxyz

where □ represents the space character.

The file IMAGE.DAT contains one or more potentially corrupted character images. A character image might be corrupted in these ways:

- at most one line might be duplicated (and the duplicate immediately follows)
- at most one line might be missing
- some '0's might be changed to '1's
- some '1's might be changed to '0's.

No character image will have both a duplicated line **and** a missing line. No more than 30% of the '0's and '1's will be changed in any character image in the evaluation datasets.

In the case of a duplicated line, one or both of the resulting lines may have corruptions, and the corruptions may be different.

Task:

Write a program to recognise the sequence of one or more characters in the image provided in file IMAGE.DAT using the font provided in file FONT.DAT.

Recognise a character image by choosing the font character images that require the smallest number of overall changed '1's and '0's to be corrupted to the given font image, given the most favourable assumptions about duplicated or omitted lines. Count corruptions in only the least corrupted line in the case of a duplicated line. All characters in the sample and evaluation images used are recognisable one-by-one by a well-written program. There is a unique best solution for each evaluation dataset.

A correct solution will use precisely all of the data supplied in the IMAGE.DAT input file.

Input:

Both input files begin with an integer N ($19 \leq N \leq 1200$) that specifies the number of lines that follow:

```
N
(digit1)(digit2)(digit3) ... (digit20)
(digit1)(digit2)(digit3) ... (digit20)
...
```

Each line of data is 20 digits wide. There are no spaces separating the zeros and ones.

The file FONT.DAT describes the font. FONT.DAT will always contain 541 lines. FONT.DAT may differ for each evaluation dataset.

Output:

Your program must produce a file IMAGE.OUT, which contains a single string of the characters recognised. Its format is a single line of ASCII text. The output should not contain any separator characters. If your program does not recognise a particular character, it must output a '?' in the appropriate position.

Caution: the output format specified above overrides the standard output requirements specified in the rules, which require separator spaces in output.

Scoring:

The score will be given as the percentage of characters correctly recognised.

SEE OTHER SIDE FOR SAMPLES.



