Hidden Password

You are given two alphanumeric ASCII strings. An ancient manuscript says those strings contain a hidden password. Decode it!

The first string may be grouped into tuples of six characters each. For each such 6-tuple, taking from the *i*-th character (start counting from 0) the *i*-th bit of its ASCII code gives you a number (call it *a*), and likewise taking the ((*i*+3) mod 6)-th bits gives you another number (call it *b*).

These two numbers tell you about the next two characters to be included in the password, namely the *a*-th and the *b*-th character from the second string (count starting from 0 as usual).

Input

First, you are given t ($t < 100$) - the number of test cases.

Each of the test cases starts with one number n ($n < 100$) - the number of 6-tuples in the first string, followed by the two strings in separate lines (please have a look at the example to see the correct format). The second string is 64 characters long.

Successive test cases are separated by an empty line.

Output

For each of the test cases, output its hidden password in a separate line.

Example

Input:

```
2
2
qwe345 rf3Arg
XSBSRasdew9873465hkldsfsalndfvnfq489uqovkLKJHaeDaae555Sk5asdpASD

3
2S4J5K 111111 lrtb2A
isimgsow45ipfgisd56wfgngdfcdkgc7kKKKkuuJJgfstdygQdWORQADFSLKF2K8
```

Output:

```
keep
coding
```

Explanation

Let us have a look at the first 6-tuple: qwe345.

```
char. ASCII code
  q   113 = 01110001B
  w   119 = 01110111B
  e   101 = 01100101B
  3    51 = 00110011B
  4    52 = 00110100B
  5    53 = 00110101B
```

a (blue bits) = 110111B = 55

b (red bits) = 101110B = 46