

A certain frazzled programmer is writing a program that receives two numbers at a time in hexadecimal form, performs an addition or subtraction on them, and outputs the result in decimal form. However, the binary representation of the hexadecimal numbers also needs to be output, in the exact format shown by the sample output below.

This programmer would gladly write the routine to do this himself, but every time he tries to do anything in base 2, he breaks out in hives. So if you write this little routine for him, he would be eternally grateful.

You may assume the following:

- The largest allowable hexadecimal number is **FFF**.
- When subtracting, the second number will always be smaller than the first, i.e. no negative results.
- The spacing in the input file will be uniform throughout, i.e. no spaces at the beginning of a line, and one space between each element.

Input

The input for this program will come from a file. The format for the file is as follows:

N (This is the number of expressions to compute)

HEX1 (+ or -) *HEX2* (The first expression)

.
.
.

HEX1 (+ or -) *HEX2* (The *n*th expression)

Output

The output file should be in the following format:

BINARY1 (+ or -) *BINARY2* = *DECIMAL* (first result)

.
.
.

BINARY1 (+ or -) *BINARY2* = *DECIMAL* (*n*th result)

Sample Input

```
2
A + 3
AAA + BBB
```

Sample Output

```
0000000001010 + 0000000000011 = 13
0101010101010 + 0101110111011 = 5733
```