Geek Challenge [SKRZAT] (Base Minus Two)

Geek Challenge [SKRZAT] is an **old**, **old** game from Poland that uses a game console with two buttons plus a joy stick. As is true to its name, the game communicates in binary, so that one button represents a zero and the other a one. Even more true to its name, the game chooses to communicate so that the base of the number system is **minus two**, not plus two, so we'll call this representation "Weird Binary". Thus the bit positions label the powers of **minus two**, as seen in the following five-bit tables:

Bits	Value	Bits	Value	Bits	Value	Bits	Value
00000	0	01000	−8	10000	16	11000	8
00001	1	01001	−7	10001	17	11001	9
00010	−2	01010	−10	10010	14	11010	6
00011	−1	01011	−9	10011	15	11011	7
00100	4	01100	−4	10100	20	11100	12
00101	5	01101	−3	10101	21	11101	13
00110	2	01110	−6	10110	18	11110	10
00111	3	01111	−5	10111	19	11111	11

Bits	Value	Bits	Value	Bits	Value	Bits	Value
01010	−10	00010	−2	11010	6	10010	14
01011	−9	00011	−1	11011	7	10011	15
01000	−8	00000	0	11000	8	10000	16
01001	−7	00001	1	11001	9	10001	17
01110	−6	00110	2	11110	10	10110	18
01111	−5	00111	3	11111	11	10111	19
01100	−4	00100	4	11100	12	10100	20
01101	−3	00101	5	11101	13	10101	21

Numbers are presented on the screen in Weird Binary, and then numbers are accepted in response from the console as a stream of zeroes and ones, terminated by a five-second pause.

You are writing a computer program to support the novice geek in playing the game by translating numbers between decimal and Weird Binary.

Input

The first line in the file gives the number of problems being posed without any white space. Following are that many lines. Each line will either be a conversion into Weird Binary or out of Weird Binary: the letter "b" indicates that the rest of the line is written in Weird Binary and needs to be converted to decimal; the letter "d" indicates that the rest of the line is written in decimal and needs to be converted to Weird Binary.

The input data are in the range to fit within a 15-bit Weird Binary number, which represents the decimal number range –10922 to 21845, inclusive.

Output

For each conversion problem, show the type of problem, its input string, and the converted result in the format shown below, replicating even the spacing exactly as shown. Leading zeroes are <u>not</u> allowed.

Sample Input

```
10
b 1001101
b 0111111
b 101001000100001
b 010010001000010
b 100110100110100
d -137
d 137
d 8191
d -10000
d 21000
```

Sample Output

```
From binary: 1001101 is 61
From binary: 0111111 is -21
From binary: 101001000100001 is 19937
From binary: 010010001000010 is -7106
From binary: 100110100110100 is 15604
From decimal: -137 is 10001011
From decimal: 137 is 110011001
From decimal: 8191 is 110000000000011
From decimal: -10000 is 10100100110000
From decimal: 21000 is 101011000011000
```