Number systems conversion

There are four number systems: decimal (10), binary (2), octal (8), and hexadecimal (16). As humans, we use the decimal system.

Name	Symbols	Notation
decimal	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	$759, 100$
binary	0, 1	$101_2, 1101_B$
octal	0, 1, 2, 3, 4, 5, 6, 7	$357_8, 156_O$
hexadecimal	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A (10), B (11), C (12), D (13), E (14), F (15)	$F5_{16}, ABC_H$

The color blue on the examples below indicates which part we write first.

When converting **dec** to **bin**, we have to write the powers of 2 from the right side to the moment the next power will be bigger than the number we want to convert. Then, going from the left, we write 1 above a value and subtract it from our main number. We continue the proces, but if a number is too large to subtract without receiving a negative number, we write 0 above it and go on. When converting **bin** to **dec**, we just invert the process and note the powers of 2 under the binary number. Then, we add the numbers that have 1 above them.

10 → 2   **2 → 10**

78 → 1001110   1001110 → 78

1 0 0 1 1 1 0   1 0 0 1 1 1 0
64 32 16 8 4 2 1   64 32 16 8 4 2 1

When converting **bin** to **hex**, we have to divide the number by a line, every four digits, and write (from the right side) the first four powers of 2 under each set of them. Then, we do a similar thing to the previous conversion, and we add the powers of 2 that have 1 above them from every set of four digits separately. Our final number is the conjunction of all the results into one string. When converting **hex** to **bin**, we just invert the process.

2 → 16   **16 → 2**

1011010 → 5A   5A → 1011010

0101 | 1010   0101 | 1010
8 4 2 1   8 4 2 1   8 4 2 1   8 4 2 1
5   10 = A   5   A

When converting **dec** to **hex**, we have to divide the number by %16 (modulo 16), and write the result on the other side of the line. We also have to write the rounded result of a normal division below the divided number, and continue the process using this number. After we finish, we read the number from the bottom to the top. When converting **hex** to **dec**, we take every digit / letter from this number (from the right side), and multiply them by succeeding powers of sixteen. When converting **dec** to **oct** and vice versa, we use the same methods, but we divide and multiply by 8.

10 → 16   **16 → 10**   **10 → 8**   **8 → 10**

42 → 2A   2A → 42   52 → 64   64 → 52

Ex.1   Ex. 2*
:16 r.   :16 r. $2A = A * 16^0 + 2 *$:8   r. $64 = 4 * 8^0 + 6 *$
42 | 10   64 | 0 $16^1 = 10 * 1 + 2 *$ 52 | 4 $8^1 = 4 * 1 + 6 * 8$
  2 | 2   4 | 4 $16 = 10 + 32 = 42$ 6 | 6 $= 4 + 48 = 52$

*This example has nothing to do with the rest of the operations.

When we convert the symbols from hexadecimal to zeros and ones, we write them on four bits (four digits), and when we convert them from octal, we write them on three bits.

When converting **bin** to **oct**, we have to do the same thing like with the conversion from **bin** to **hex**, but we divide the number with lines every **three** digits.

2 → 8   **8 → 2**

1011010 → 132   132 → 1011010

001 | 011 | 010   001 | 011 | 010
4 2 1   4 2 1   4 2 1   4 2 1   4 2 1   4 2 1
 1   3   2   1   3   2

Conversion directly from **hex** to **oct**, and vice versa, is **impossible**. We have to do it indirectly through binary.

Octal	Binary
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Hexadecimal	Binary
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111