Number systems

Binary

We are using decimal number system, that is the amount of digits before counting to 10 is 10.
`0, 1, 2, 3, 4, 5, 6, 7, 8, 9`
What if instead of using 10 digits, we were using just two? We get a binary number system.
`0, 1, 10, 11, 100, 101, 110, 111, 1000, 1001`

To distinguish between number systems, we can specify the amount of digits under the number (subscripting):

For decimal: 123_{10} For binary: 100_{2}

The reasons for using and studying binary system is electricity and computers. Either we have power on = `1` , or we have it off = `0`

Quests:

👍

Convert to binary the numbers from 1 to 32

🏅

Convert to binary the sentence: `Binary is the native language of computers`

Homework:

👍

Research how binary system was invented

Decimal to Binary

Write number, if it is even, write 0 to the right, else write 1.
Write number divided by two without remainder, write 0 or 1 if it is even or odd.
Till number is equal to 1.
Write binary from bottom to top.

```
77 1  46 0  123 1    1. 77  = 1001101
38 0  23 1   61 1    2. 46  = 101110
19 1  11 1   30 0    3. 123 = 1111011
 9 1  5  1   15 1
 4 0  2  0    7 1
 2 0     1    3 1
   1            1
```

Quests:

👍

Convert any 5 decimals greater than 100

🎖️

Convert any 2 decimals greater than 10 000 000

Binary to Decimal

```
2^0 = 1    2^1 = 2    2^2  = 2 * 2 = 4    2^3  = 2 * 2 * 2 = 8
2^4 = 16   2^5 = 32   2^6  = 64           2^7  = 128
2^8 = 256  2^9 = 512  2^10 = 1024         2^11 = 2048
```

Each binary, as any number, is made of digits; individual digit in binary is called bit. Bit is either `1` or `0`. So number 10001 has five bits, two of which are ones.

If bit is `1`, then some power of two is present in the sum of its decimal counterpart.
The exact power of two is determined by the position of bit in the number.
The position is determined by counting from 0 and Starting from right to left
`10001 = 2^0 + 2 ^ 4 = 17`

```
101 = 2^0 + 2^2 = 1 + 4
1000 = 2^3 = 8
111 = 2^0 + 2^1 + 2^2 = 1 + 2 + 4 = 7

10101010
^^^^^^^^
76543210 = 2^7 + 2^5 + 2^3 + 2^1 = 128 + 32 + 8 + 2 = 128 + 42 = 170
```

Quests

Convert any 5 binaries with digits less than 6 into decimal

Convert any 5 binaries with digits greater than 8 into decimal

Decimal to Binary additional

By using bintodec algorithm in reverse, we can translate from decimal to binary

```
777 = 512 + 265
777 = 512 + 256 + 9
777 = 512 + 256 + 8 + 1
powers of two:
9830 = 1100001001
```

Quests

👍

Convert any 5 decimals with digits less than 4 into binary

🎖️

Convert any 3 decimals with digits greater than 3 into decimal

Hexadecimal

When working with binary numbers, it is too difficult to make sense of them because of too many digits. To resolve this inconvenience, a hexadecimal (hex) number system is used.

From Greek hex - "six", and latin decimal - "10", this number system contains 16 digits
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Every four bits can be represented with just one digit in hex.

```
0001 = 1    0010 = 2    0011 = 3    0100 = 4
0101 = 5    0110 = 6    0111 = 7    1000 = 8
1001 = 9    1010 = A    1011 = B    1100 = C
1101 = D    1110 = E    1111 = F    10000 = 10

1010 1100 0011 = AC3
```

This number system is heavily used to encode colors.

Quests

👍

Convert the following to hex:
`1001111000100111`
`1001100000000000`
`0110101111010111`
`1000010011011110`

🔑

Convert this to hex:
10011110001001111001100000000000011010111101011110000100110111101001111000100111100110
0000000000011010111101011110000100110111101001111000100111100110000000000001101011110
10111100001001101111010011110001001111001100000000000001101011110101111000010011011110 1
0011110001001111001100000000000011010111101011110000100110111101001111000100111 1001100
00000000011010111101011110000100110111101001111000100111100110000000000000 11010111101
01111000010011011110100111100010011110011000000000000011010111101011110000100110111101 0
01111000100111100110000000000000110101111010111100001001101111010011110001001111 0011000
000000000110101111010111100001001101111 0

(Hint: Look for any patterns ~~16~~ ~~10~~)