Day 10: Binary Numbers | HackerRank

Terms you'll find helpful in completing today's challenge are outlined below, along with sample Java code (where appropriate).

Radix (Base)

The number of digits that can be used to represent a number in a positional number system. The decimal number system (base-10) has 10 digits ($0, 1, 2, 3, 4, 5, 6, 7, 8, 9$); the binary (base-$2$) number system has 2 digits ($0, 1$).

We think in terms of base-10, because the decimal number system is the only one many people need in everyday life. For situations where there is a need to specify a number's radix, number n having radix r should be written as $(n)_r$.

Binary to Decimal Conversion

In the same way that
$(840)_{10} = (8 \times 10^2) + (4 \times 10^1) + (0 \times 10^0) = 800 + 40 + 0 = 840$, a binary number having k digits in the form of $d_{k-1}d_{k-2}\ldots d_2d_1d_0$ can be converted to decimal by summing the result for each $d_i \times 2^i$ where $0 \leq i \leq k-1$, $i = k-1$ is the most significant bit, and $i = 0$ is the least significant bit.

For example: $(1011)_2 \rightarrow (?)_{10}$ is evaluated as
$(1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) = 8 + 0 + 2 + 1 = (11)_{10}$

Decimal to Binary Conversion

To convert an integer from decimal to binary, repeatedly divide your base-10 number, n, by 2. The dividend at each step i should be the result of the integer division at each step $i-1$. The remainder at each step of division is a single digit of the binary equivalent of n; if you then read each remainder in order from the last remainder to the first (demonstrated below), you have the entire binary number.

For example: $(4)_{10} \rightarrow (?)_2$. After performing the steps outlined in the above paragraph, the remainders form $(100)_2$ (the binary equivalent of $(4)_{10}$) when read from the bottom up:

This can be expressed in pseudocode as:

```
while(n > 0):
    remainder = n%2;
    n = n/2;
    Insert remainder to front of a list or push onto a stack
```

Print list or stack

Many languages have built-in functions for converting numbers from decimal to binary. To convert an integer, n, from decimal to a String of binary numbers in Java, you can use the *Integer.toBinaryString(n)* function.

Note: The algorithm discussed here is for converting integers; converting fractional numbers is a similar (but different) process.

Solve Problem