

COUNTING BITS

Given an integer, n, we want to know the following:

1. How many 1-bits are in its binary representation?
2. Let's say n's binary representation has k significant bits indexed from 1 to k. What are the respective positions (i.e., in ascending order) of each 1-bit?
3. The performance is really important in this challenge.

Example:

Complete Count function in PositiveBitCounter class. It has one parameter: an integer, n. It must return an integer enumerable with the following 1 + k values:

The first index (0) must contain the total number of 1 bits in n's binary representation.

The subsequent indices must contain the respective positions of the one-indexed 1-bits in n's binary representation.

Output Format:

Return an enumerable of integers where the first element is the total number of 1-bits in n's binary representation and the subsequent elements are the respective one-indexed locations of each 1-bit from most to least significant.

Tips:

The integer n = 161 converts to binary.

1	0	1	0	0	0	0	1
---	---	---	---	---	---	---	---

Reverse the binary representation.

1	0	0	0	0	1	0	1
---	---	---	---	---	---	---	---

Count number of positive bits: 3

Search the position: 0, 5, 7

Return { 3, 0, 5, 7 }