Problem 191: Number of 1 Bits

Problem Information

Difficulty: Easy
Acceptance Rate: 75.69%
Paid Only: No
Tags: Divide and Conquer, Bit Manipulation

Problem Description

Given a positive integer `n`, write a function that returns the number of set bits in its binary representation (also known as the [Hamming weight](http://en.wikipedia.org/wiki/Hamming_weight)).

Example 1:

Input: n = 11

Output: 3

Explanation:

The input binary string **1011** has a total of three set bits.

Example 2:

Input: n = 128

Output: 1

Explanation:

The input binary string **10000000** has a total of one set bit.

Example 3:

Input: n = 2147483645

Output: 30

Explanation:

The input binary string **1111111111111111111111111111101** has a total of thirty set bits.

Constraints:

* `1 <= n <= 231 - 1`

Follow up: If this function is called many times, how would you optimize it?

Code Snippets

C++:

```
class Solution {
public:
int hammingWeight(int n) {

}
};
```

Java:

```
class Solution {
public int hammingWeight(int n) {

}
}
```

Python3:

```
class Solution:
def hammingWeight(self, n: int) -> int:
```