Write a function that takes an unsigned integer and returns the number of '1' bits it has (also known as the [Hamming weight](http://en.wikipedia.org/wiki/Hamming_weight)).

**Note:**

* Note that in some languages, such as Java, there is no unsigned integer type. In this case, the input will be given as a signed integer type. It should not affect your implementation, as the integer's internal binary representation is the same, whether it is signed or unsigned.
* In Java, the compiler represents the signed integers using [2's complement notation](https://en.wikipedia.org/wiki/Two%27s_complement). Therefore, in **Example 3**, the input represents the signed integer. -3.

 public class Solution {

// you need to treat n as an unsigned value

public int hammingWeight(int n) {

int result=0, r=n&0x80000000;

if(r!=0) result++;

n=n&0x7FFFFFFF;

while(n!=0)

{

int rem=n%2;

if(rem==1) result++;

n=n/2;

}

return result;

}

}

