Position of rightmost set bit

Write a one line function to return position of first 1 from right to left, in binary representation of an Integer.

```
I/P    18,   Binary Representation 010010

O/P   2

I/P    19,   Binary Representation 010011

O/P   1
```

Algorithm: (Example **12(1100)**)

Let I/P be 12 (1100)

1. Take two's complement of the given no as all bits are reverted

except the first '1' from right to left (0100)

2 Do a bit-wise & with original no, this will return no with the

required one only (0100)

3 Take the log2 of the no, you will get (position – 1) (2)

4 Add 1 (3)

Explanation –

(n&~(n-1)) always return the binary number containing rightmost set bit as 1.

if N = 12 (1100) then it will return 4 (100)

Here log2 will return you, number of times we can express that number in power of two.

For all binary number containing **only** rightmost set bit as 1 like 2 , 4, 8, 16, 32….

We will find that position of right most set bit is always equals to log2(Number)+1