PROBLEM

Reverse Bits🔖

Easy   **Accuracy: 53.96%**   **Submissions: 50K+**   **Points: 2**

Given a **number x**, **reverse** its binary form and return the answer in **decimal**.

Example 1:

```
Input:
x = 1
Output:
2147483648
Explanation:
Binary of 1 in 32 bits representation-
00000000000000000000000000000001
Reversing the binary form we get,
10000000000000000000000000000000,
whose decimal value is 2147483648.
```

Example 2:

```
Input:
x = 5
Output:
2684354560
Explanation:
Binary of 5 in 32 bits representation-
00000000000000000000000000000101
Reversing the binary form we get,
10100000000000000000000000000000,
whose decimal value is 2684354560.
```

Your Task:

You don't need to read input or print anything. Your task is to complete the function **reversedBits()** which takes an Integer **x** as input and returns the reverse binary form of **x** in decimal form.

Expected Time Complexity: $O(\log(x))$
Expected Auxiliary Space: $O(1)$

Constraints:
$0 \le x < 2^{32}$

CODE

```python
#User function Template for python3


class Solution:

    def reversedBits(self, x):

        # code here

        res = 0


        for i in range(0,32):

            res = res << 1


            bit = x%2


            res+=bit


            x = x >> 1


        return res
```