Reverse the order of the bits in a given integer.

**Example**

* For a = 97, the output should be  
  mirrorBits(a) = 67.

97 equals to 1100001 in binary, which is 1000011 after mirroring, and that is 67 in base 10.

* For a = 8, the output should be  
  mirrorBits(a) = 1.

**Input/Output**

* **[time limit] 3000ms (cs)**
* **[input] integer a**

*Constraints:*  
5 ≤ a ≤ 105.

* **[output] integer**

<https://codefights.com/arcade/code-arcade/corner-of-0s-and-1s/e3zfPNTwTa9qTQzcX>

public static int toDec(String bin)

{

char[] temporal = bin.ToCharArray();

Array.Reverse(temporal);

int NumeroDecimal = 0;

for (int i = bin.Length - 1; i >= 0; i--)

{

NumeroDecimal += Convert.ToInt32(int.Parse(temporal[i].ToString()) \* Math.Pow(2, i));

}

return NumeroDecimal;

}

static int mirrorBits(int a)

{

string bin = "";

while (a > 0)

{

bin += (a % 2).ToString();

a /= 2;

}

return toDec(bin);

}