We have two arrays arr1 and arr2 which are initially empty. You need to add positive integers to them such that they satisfy all the following conditions:

* arr1 contains uniqueCnt1 **distinct** positive integers, each of which is **not divisible** by divisor1.
* arr2 contains uniqueCnt2 **distinct** positive integers, each of which is **not divisible** by divisor2.
* **No** integer is present in both arr1 and arr2.

Given divisor1, divisor2, uniqueCnt1, and uniqueCnt2, return *the* ***minimum possible maximum*** *integer that can be present in either array*.

**Example 1:**

Input: divisor1 = 2, divisor2 = 7, uniqueCnt1 = 1, uniqueCnt2 = 3  
Output: 4  
Explanation:   
We can distribute the first 4 natural numbers into arr1 and arr2.  
arr1 = [1] and arr2 = [2,3,4].  
We can see that both arrays satisfy all the conditions.  
Since the maximum value is 4, we return it.

**Example 2:**

Input: divisor1 = 3, divisor2 = 5, uniqueCnt1 = 2, uniqueCnt2 = 1  
Output: 3  
Explanation:   
Here arr1 = [1,2], and arr2 = [3] satisfy all conditions.  
Since the maximum value is 3, we return it.

**Example 3:**

Input: divisor1 = 2, divisor2 = 4, uniqueCnt1 = 8, uniqueCnt2 = 2  
Output: 15  
Explanation:   
Here, the final possible arrays can be arr1 = [1,3,5,7,9,11,13,15], and arr2 = [2,6].  
It can be shown that it is not possible to obtain a lower maximum satisfying all conditions.

**Constraints:**

* 2 <= divisor1, divisor2 <= 105
* 1 <= uniqueCnt1, uniqueCnt2 < 109
* 2 <= uniqueCnt1 + uniqueCnt2 <= 109