Given a divisor and a bound, find the largest integer N such that:

* N is divisible by divisor.
* N is less than or equal to bound.
* N is greater than 0.

It is guaranteed that such a number exists.

**Example**

For divisor = 3 and bound = 10, the output should be  
maxMultiple(divisor, bound) = 9.

The largest integer divisible by 3 and not larger than 10 is 9.

**Input/Output**

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

*Constraints:*  
2 ≤ divisor ≤ 10.

* **[input] integer bound**

*Constraints:*  
5 ≤ bound ≤ 100.

* **[output] integer**

The largest integer not greater than boundthat is divisible by divisor.

<https://codefights.com/arcade/code-arcade/intro-gates/HEsmEacHr2s9wahjr>

static int maxMultiple(int divisor, int bound)

{

int ans = 0;

for (int n = bound; n >= 1; n--)

{

if (n % divisor == 0)

{

ans = n;

break;

}

}

return ans;

}