

In bit-wise expression, mask is a common term. You can get a certain bit-pattern using mask. For example, if you want to make first 4 bits of a 32-bit number zero, you can use `0xFFFFFFFF0` as mask and perform a bit-wise AND operation. Here you have to find such a bit-mask.

Consider you are given a 32-bit unsigned integer N . You have to find a mask M such that $L \leq M \leq U$ and $N \text{ OR } M$ is maximum. For example, if N is 100 and $L = 50$, $U = 60$ then M will be 59 and $N \text{ OR } M$ will be 127 which is maximum. If several value of M satisfies the same criteria then you have to print the minimum value of M .

Input

Each input starts with 3 unsigned integers N , L , U where $L \leq U$. Input is terminated by EOF.

Output

For each input, print in a line the minimum value of M , which makes $N \text{ OR } M$ maximum.

Look, a brute force solution may not end within the time limit.

Sample Input

```
100 50 60
100 50 50
100 0 100
1 0 100
15 1 15
```

Sample Output

```
59
50
27
100
1
```