

2개 이하로 다른 비트

Python3

문제 설명

양의 정수 x 에 대한 함수 $f(x)$ 를 다음과 같이 정의합니다.

- x 보다 크고 x 와 비트가 1~2개 다른 수들 중에서 제일 작은 수

예를 들어,

- $f(2) = 3$ 입니다. 다음 표와 같이 2보다 큰 수들 중에서 비트가 다른 지점이 2개 이하이면서 제일 작은 수가 3이기 때문입니다.

수	비트	다른 비트의 개수
2	000...0010	
3	000...0011	1

- $f(7) = 11$ 입니다. 다음 표와 같이 7보다 큰 수들 중에서 비트가 다른 지점이 2개 이하이면서 제일 작은 수가 11이기 때문입니다.

수	비트	다른 비트의 개수
7	000...0111	
8	000...1000	4
9	000...1001	3
10	000...1010	3
11	000...1011	2

1씩 더해보면 항상 2개 이상 비트가 다른 수를 찾는다.
 → 7(0111)은 2진수는 0으로 끝나는 정수에 4(100)이 더해진 것.
 → 7(0111)은 2진수는 0으로 끝나는 정수에 2(10)이 더해진 것.
 0이 없으므로 1씩 더해보면 2개 이상 비트가 다른 수를 찾는다.

정수들이 담긴 배열 `numbers`가 매개변수로 주어집니다. `numbers`의 모든 수들에 대하여 각 수의 f 값을 배열에 차례대로 담아 return 하도록 solution 함수를 완성해주세요.

제한사항

- $1 \leq \text{numbers}$ 의 길이 $\leq 100,000$
- $0 \leq \text{numbers}$ 의 모든 수 $\leq 10^{15}$

입출력 예

numbers	result
[2, 7]	[3, 11]

입출력 예 설명

입출력 예 #1

- 문제 예시와 같습니다.

solution.py

```
1 def solution(numbers):
2     answer = []
3     return answer
```

실행 결과

실행 결과가 여기에 표시됩니다.

- 초기화
- 코드 실행
- 제출 후 채점하기