

Problem A. La pirámide

Time limit 2000 ms

Mem limit 262144 kB

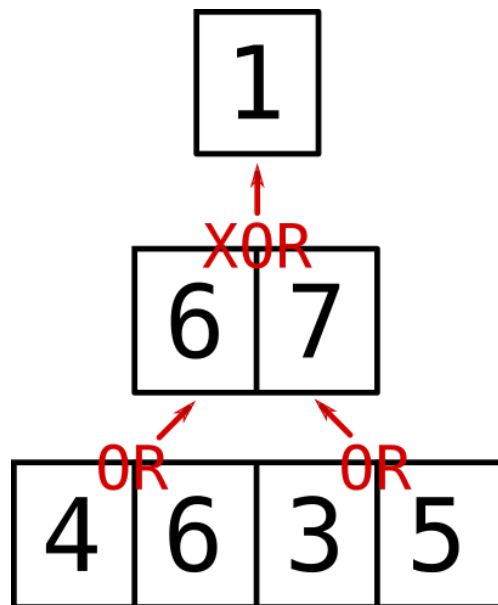
La pirámide está de vuelta. Esta vez, tendremos un arreglo a de 2^n enteros no-negativos a_1, a_2, \dots, a_{2^n} . A partir de este arreglo a , podemos formar una pirámide.

La pirámide se construye en varias iteraciones, desde abajo hacia arriba. En la base de la pirámide escribiremos el arreglo a original, es decir, a_1, a_2, \dots, a_{2^n} .

En la primera iteración, escribiremos el arreglo a_1 OR a_2 , a_3 OR a_4 , \dots , $a_{2^{n-1}}$ OR a_{2^n} arriba. Es decir, calculamos el **OR** binario entre elementos adyacentes del arreglo.

En la segunda iteración, haremos lo mismo pero con el **OR** exclusivo (**XOR**) de los elementos adyacentes, en la tercera iteración con el **OR**, en la cuarta con el **XOR**, y así alternadamente hasta llegar a un solo elemento: la punta de la pirámide.

A continuación puedes ver cómo quedaría una pirámide empezando con el arreglo 4, 6, 3, 5:



Te entregaremos el arreglo inicial a (la base de la pirámide). Como calcular la punta de esta pirámide sería muy fácil, te haremos m consultas. Cada consulta consiste en un par de números enteros (i, x) , y significa que debes asignar $a_i = x$. Después de cada consulta, debes imprimir el valor de la punta de la pirámide que tiene al arreglo a como base.

Entrada

La primera línea contiene dos enteros n y m ($1 \leq n \leq 17, 1 \leq m \leq 10^5$). La siguiente línea contiene 2^n enteros a_1, a_2, \dots, a_{2^n} ($0 \leq a_i \leq 2^{30}$).

Cada una de las siguientes m líneas contienen una consulta. Cada consulta consiste en un par de enteros (i, x) separados por espacios ($1 \leq i \leq 2^n, 0 \leq x \leq 2^{30}$).

Salida

Imprime una línea por cada consulta, indicando el valor de la punta de la pirámide después de la actualización.

Ejemplos

Input
2 4 2 6 3 5 1 4 3 4 1 2 1 2
Output
1 3 3 3

Nota: el arreglo después de la primera actualización corresponde al ejemplo del enunciado.