Problem Link:

<https://leetcode.com/problems/smallest-subarrays-with-maximum-bitwise-or/description/?envType=daily-question&envId=2025-07-29>

Solution:

class Solution {

public:

vector<int> smallestSubarrays(vector<int>& nums) {

int n = nums.size();

vector<int> v(n);

map<int, int> m;

for(int i = n - 1; i >= 0; --i)

{

map<int, int> nm;

nm[nums[i]] = 1;

for(auto& p : m)

{

int no = p.first | nums[i];

if(nm.count(no))

{

nm[no] = min(nm[no], p.second + 1);

}

else

{

nm[no] = p.second + 1;

}

}

m = move(nm);

int mx = 0;

for(auto& p : m)

{

mx = max(mx, p.first);

}

v[i] = m[mx];

}

return v;

}

};