You are given a **(0-indexed)** array of positive integers candiesCount where candiesCount[i] represents the number of candies of the ith type you have. You are also given a 2D array queries where queries[i] = [favoriteTypei, favoriteDayi, dailyCapi].

You play a game with the following rules:

* You start eating candies on day **0**.
* You **cannot** eat **any** candy of type i unless you have eaten **all** candies of type i - 1.
* You must eat **at least** **one** candy per day until you have eaten all the candies.

Construct a boolean array answer such that answer.length == queries.length and answer[i] is true if you can eat a candy of type favoriteTypei on day favoriteDayi without eating **more than** dailyCapi candies on **any** day, and false otherwise. Note that you can eat different types of candy on the same day, provided that you follow rule 2.

Return *the constructed array* answer.

**Example 1:**

Input: candiesCount = [7,4,5,3,8], queries = [[0,2,2],[4,2,4],[2,13,1000000000]]  
Output: [true,false,true]  
Explanation:  
1- If you eat 2 candies (type 0) on day 0 and 2 candies (type 0) on day 1, you will eat a candy of type 0 on day 2.  
2- You can eat at most 4 candies each day.  
 If you eat 4 candies every day, you will eat 4 candies (type 0) on day 0 and 4 candies (type 0 and type 1) on day 1.  
 On day 2, you can only eat 4 candies (type 1 and type 2), so you cannot eat a candy of type 4 on day 2.  
3- If you eat 1 candy each day, you will eat a candy of type 2 on day 13.

**Example 2:**

Input: candiesCount = [5,2,6,4,1], queries = [[3,1,2],[4,10,3],[3,10,100],[4,100,30],[1,3,1]]  
Output: [false,true,true,false,false]

**Constraints:**

* 1 <= candiesCount.length <= 105
* 1 <= candiesCount[i] <= 105
* 1 <= queries.length <= 105
* queries[i].length == 3
* 0 <= favoriteTypei < candiesCount.length
* 0 <= favoriteDayi <= 109
* 1 <= dailyCapi <= 109