1200. Relative Ranks

Given scores of N athletes, find their relative ranks and the people with the top three highest scores, who will be awarded medals: "Gold Medal", "Silver Medal" and "Bronze Medal".

Example

Example 1:

Input: [5, 4, 3, 2, 1]

Output: ["Gold Medal", "Silver Medal", "Bronze Medal", "4", "5"]

Explanation: The first three athletes got the top three highest scores, so they got "Gold Medal", "Silver Medal" and "Bronze Medal".

For the right two athletes, you just need to output their relative ranks according to their scores.

Notice

N is a positive integer and won't exceed 10,000.  
All the scores of athletes are guaranteed to be unique.

<https://www.lintcode.com/problem/relative-ranks/description>

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication68;

import java.util.ArrayList;

import java.util.LinkedList;

import java.util.Queue;

/\*\*

\*

\* @author Usuario

\*/

public class JavaApplication68 {

public static String[] findRelativeRanks(int[] nums) {

// write your code here

//nums = CountSort(nums);

String[] result = new String[nums.length];

int max = -1; //nums.Max();

for(int i =0; i<nums.length; i++) {

max = Math.max(max, nums[i]);

}

int[] hash = new int[max + 1];

for (int i = 0; i < nums.length; i++)

hash[nums[i]] = i + 1;

int place = 1;

for (int i = hash.length - 1; i >= 0; i--)

{

if (hash[i] != 0)

{

if (place == 1)

result[hash[i] - 1] = "Gold Medal";

else if (place == 2)

result[hash[i] - 1] = "Silver Medal";

else if (place == 3)

result[hash[i] - 1] = "Bronze Medal";

else

result[hash[i] - 1] = String.valueOf( place);

place++;

}

}

return result;

}

public static void main(String[] args) {

// TODO code application logic here

//int[] a = {5,4,3,2,1 };

int[] a = {1 };

//int[] a = { 1 };

String[] res = findRelativeRanks(a);

for(int i =0; i<res.length; i++) {

System.out.print(res[i] + " ");

}

}

}