Minimum Unique Array Sum

Given an array, increment any duplicate elements until all elements are unique. The sum of the elements must be the minimum possible.

**Example**

*arr* = *[3, 2, 1, 2, 7]*

The minimum *arrunique* = *[3, 2, 1, 4, 7]* and its elements sum to the minimal value of *3 + 2 + 1 + 4 + 7 = 17*.

**Function Description**

Complete the *getMinimumUniqueSum* function in the editor below.

*getMinimumUniqueSum* has the following parameter(s):

*int arr[n]:*  an array of integers to process

Returns:

*int:* an integer, thesum of the resulting array

**Constraints**

* 1 ≤ *n* ≤ 2000
* 1 ≤ *arr[i]* ≤ 3000 where 0 ≤ *i* < *n*

Input Format For Custom Testing

Input from stdin will be processed as follows and passed to the function:

The first line contains an integer, *n*, the size of array *arr*.

Each of the next *n* lines contains an integer that describes element *arr[i]* where 0 ≤ *i* < *n.*

Sample Case 0

**Sample Input**

STDIN Function

----- -----

3 → arr[i] size n = 3

1 → arr = [1, 2, 2]

2

2

**Sample Output**

6

**Explanation**

The duplicate array elements *2* must be addressed. Increment one of the twos by *1*, resulting in *arrunique =[1, 2, 3]*. The sum of elements in the new array is *1 + 2 + 3 = 6*.

Sample Case 1

**Sample Input**

STDIN Function

----- -----

3 → arr[i] size n = 3

1 → arr = [1, 2, 3]

2

3

**Sample Output**

6

**Explanation**

Each number in *arr* is unique, so there is no need to modify its elements (i.e., *arr ≡ arrunique*). The sum of all elements in the array is *1 + 2 + 3 = 6*.

Sample Case 2

**Sample Input**

STDIN Function

----- -----

4 → arr[i] size n = 4

2 → arr = [2, 2, 4, 5]

2

4

5

**Sample Output**

14

**Explanation**

Elements *arr[0]* and *arr[1]* are duplicates, so increment one of them to *3*. The new array, *arrunique = [2, 3, 4, 5]*.

The sum of these elements is *2 + 3 + 4 + 5 = 14*.

import java.io.\*;

import java.math.\*;

import java.security.\*;

import java.text.\*;

import java.util.\*;

import java.util.concurrent.\*;

import java.util.function.\*;

import java.util.regex.\*;

import java.util.stream.\*;

import static java.util.stream.Collectors.joining;

import static java.util.stream.Collectors.toList;

class Result {

/\*

\* Complete the 'getMinimumUniqueSum' function below.

\*

\* The function is expected to return an INTEGER.

\* The function accepts INTEGER\_ARRAY arr as parameter.

\*/

public static int getMinimumUniqueSum(List<Integer> arr) {

// Write your code here

}

}

public class Solution {

public static void main(String[] args) throws IOException {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT\_PATH")));

int arrCount = Integer.parseInt(bufferedReader.readLine().trim());

List<Integer> arr = IntStream.range(0, arrCount).mapToObj(i -> {

try {

return bufferedReader.readLine().replaceAll("\\s+$", "");

} catch (IOException ex) {

throw new RuntimeException(ex);

}

})

.map(String::trim)

.map(Integer::parseInt)

.collect(toList());

int result = Result.getMinimumUniqueSum(arr);

bufferedWriter.write(String.valueOf(result));

bufferedWriter.newLine();

bufferedReader.close();

bufferedWriter.close();

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*