493. Reverse Pairs

Description | Submission | Solutions

- Total Accepted: **2410**
- Total Submissions: **13513**
- Difficulty: **Hard**
- Contributors: **ckcz123**

Given an array `nums`, we call `(i, j)` an ***important reverse pair*** if `i < j` and `nums[i] > 2*nums[j]`.

You need to return the number of important reverse pairs in the given array.

Example1:

```
Input: [1,3,2,3,1]

Output: 2
```

Example2:

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

Output: 3
```

Note:

1. The length of the given array will not exceed `50,000`.
2. All the numbers in the input array are in the range of 32-bit integer.

```
#include<iostream>
#include<stdio.h>
#include<algorithm>
#include<unordered_set>
#include<string>
using namespace std;
// cancel continuous chars with nums >= 3
```

```
void mergeSort(vector<int> &nums, int left, int right, int& ret)
{
    if(left>=right) return;
    int mid = left + (right-left)/2;
    mergeSort(nums,left,mid,ret);
    mergeSort(nums,mid+1,right,ret);

    //count
    int counter = 0;
    for(int l=left,r=mid+1;l<=mid ;)
    {
        if(r>right || ((long)nums[l] <= 2*nums[r]))
        {
            l++;
            ret+=counter;
        }else
        {
            r++;
            counter++;
        }
    }

    //sort
    int temp[right-left+1];
    int k=0;
    for(int l=left,r=mid+1;l<=mid || r<=right;)
    {
        if(l<=mid && (r>right || nums[l]<nums[r]))
        {
            temp[k++] = nums[l++];
        }else{
            temp[k++] = nums[r++];
        }
    }
    //copy
    for(int i=0;i<right-left+1;++i) nums[left+i]=temp[i];
}

int main(int argc,char *argv[])
{
    vector<int> test = {1,3,2,3,1};
    int ret = 0;
```

```
    mergeSort(test,0,4,ret);
    //for (int i=0;i<(int)test.size();i++) cout<< test[i] << " ";
    printf("\n");
    cout<<ret;
    return 0;
}
```