**算法训练 删除数组零元素**

时间限制：1.0s   内存限制：512.0MB

问题描述

从键盘读入n个整数放入数组中，编写函数CompactIntegers，删除数组中所有值为0的元素，其后元素向数组首端移动。注意，CompactIntegers函数需要接受数组及其元素个数作为参数，函数返回值应为删除操作执行后数组的新元素个数。输出删除后数组中元素的个数并依次输出数组元素。

样例输入: （输入格式说明：5为输入数据的个数，3 4 0 0 2 是以空格隔开的5个整数）  
5   
3 4 0 0 2  
样例输出:（输出格式说明：3为非零数据的个数，3 4 2 是以空格隔开的3个非零整数）  
3  
3 4 2

样例输入:   
7  
0 0 7 0 0 9 0  
样例输出:  
2  
7 9

样例输入:   
3  
0 0 0  
样例输出:  
0

本题的C++参考代码如下：

#include "iostream"

using namespace std;

int main()

{

int n;

int \*arr;

cin>>n;

arr = new int[n];

int num = 0;

for (int i=0; i<n; ++i)

{

cin>>arr[i];

if(arr[i] != 0)

{

++num;

}

}

cout<<num<<endl;

for (int i=0; i<n; ++i)

{

if(arr[i] == 0)

{

continue;

}

cout<<arr[i]<<" ";

}

return 0;

}

本题的C参考代码如下：

#include "stdio.h"

int CompactIntegers(int a[], int len)

{

int i, j, k;

for(k=0; k<len; k++)

for(i=0; i<len; i++)

{

if(a[i] == 0)

{

for(j=i; j<len-1; j++)

a[j] = a[j+1];

len--;

}

}

return len;

}

void print(int a[], int len)

{

int i;

for(i=0; i<len; i++)

printf("%d ", a[i]);

printf("\n");

}

int main()

{

int a[100000];

int n;

scanf("%d", &n);

int i;

for(i=0; i<n; i++)

scanf("%d", &a[i]);

int len = CompactIntegers(a, n);

if(len>1)

{

printf("%d\n", len);

print(a, len);

}

else

printf("%d", 0);

getchar();

getchar();

getchar();

return 0;

}

本题的Java参考代码如下：

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class Main {

public void input() {

Scanner cin =new Scanner(System.in);

int n = cin.nextInt();

int [] num =new int[n];

int [] num1 =new int[n];

int j=0;

for(int i=0;i<n;i++){

num1[i]=cin.nextInt();

if(num1[i]!=0){

num[j++]=num1[i];

}

}

System.out.println(j);

if(j>0){

for(int i=0;i<j;i++)

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

}

System.out.println();

}

public static void main(String[] args) {

new Main().input();

}

}