**Segregate Even and Odd numbers**

[array](http://www.practice.geeksforgeeks.org/tag-page.php?tag=array&isCmp=0)

**Problem Description:**  
Given an array A[], write a program that segregates even and odd numbers. The program should put all even numbers first in sorted order, and then odd numbers in sorted order.

**Input:**  
The first line of input contains an integer T denoting the number of test cases. The description of T test cases follows.  
The first line of each test case contains a single integer N denoting the size of array.  
The second line contains N space-separated integers A1, A2, ..., AN denoting the elements of the array.

**Output:**  
Print the segregated array.

**Constraints:**  
1 ≤ T ≤ 100  
1 ≤ N ≤ 100  
0 ≤A[i]<=1000

**Example:**  
Input:  
2  
7  
12 34 45 9 8 90 3  
5  
0 1 2 3 4  
Output:  
8 12 34 90 3 9 45  
0 2 4 1 3

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=488>

#include <iostream>

#include <stdio.h>

#include <vector>

#include <algorithm>

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

int n;

scanf("%d", &n);

int arr[n];

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

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

}

std::vector<int> pares;

std::vector<int> impares;

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

if(arr[i] %2==0) {

pares.push\_back(arr[i]);

} else {

impares.push\_back(arr[i]);

}

}

std::sort(pares.begin() , pares.end());

std::sort(impares.begin(), impares.end());

for(int i =0; i<pares.size(); i++) {

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

}

for(int i=0; i<impares.size(); i++) {

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

}

printf("\n");

}

return 0;

}