[B - Largest Concatenated Number](https://vjudge.net/problem/HackerRank-si-largest-concatenated-number" \t "_blank)

 Given an array of integers, find the largest number that can be constructed by concatenating all the elements of the given array.

**Input Format**

First line of input contains T - number of test cases. Its followed by 2T lines. First line of each test case contains N - size of the array and the second line contains N integers - elements of the array.

**Constraints**

1 <= T <= 1000  
1 <= N <= 1000  
0 <= ar[i] <= 1000

**Output Format**

For each test case, print the largest number that can be constructed by concatenating all the elements of the given array, separated by newline.

**Sample Input 0**

3

8

49 73 58 30 72 44 78 23

4

69 9 57 60

2

40 4

**Sample Output 0**

7873725849443023

9696057

440

**Explanation 0**

Self Explanatory

#include <string>

#include <iostream>

#include <vector>

#include <algorithm>

using namespace *std*;

bool compareS(int a, int b)

{

*string* ab = *to\_string*(a) + *to\_string*(b);

*string* ba = *to\_string*(b) + *to\_string*(a);

if (ab > ba) return true;

return false;

}

*string* largestNumber(*vector*<int>& nums) {

*sort*(nums.*begin*(), nums.*end*(), compareS);

*string* max;

for (const auto& i : nums) {

max.*append*(*to\_string*(i));

}

if (!max.*empty*() && max[0] == '0') {

return "0";

}

return max;

}

int main(void)

{

int t; *cin* >> t;

while (t--)

{

int n; *cin* >> n;

*vector*<int> arr;

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

{

int ele; *cin* >> ele;

arr.*push\_back*(ele);

}

*cout* << largestNumber(arr) << "\n";

}

return 0;

}