**Form largest number from digits**

Submissions: [4269](https://practice.geeksforgeeks.org/problem_submissions.php?pid=1703)  Accuracy:

42.85%

   Difficulty: [Basic](https://practice.geeksforgeeks.org/Basic/0/0/)   Marks: 1

Show Topic Tags   

[Paytm](https://practice.geeksforgeeks.org/company/Paytm/)

Given an array of numbers form **0 to 9** of size **N**. Your task is to rearrange elements of the array such that after combining all the elements of the array number formed is maximum.

**Input:**  
The first line of input contains an integer **T** denoting the number of test cases. Then T test cases follow. The first line of each test case contains an integer N denoting the number of elements in the array. Then in the next line are N space seperated integers denoting the elements of the array.  
  
**Output:**  
For each test case print a single line a number denoting the largest number that can be achieved by rearranging the elements of the array.  
  
**Constraints:**  
1 <= T <= 103  
1 <= N <= 107  
0 <= Ai<= 9

**Example:**  
**Input:**  
2  
5  
9 0 1 3 0  
3  
1 2 3

**Output:**  
93100  
321

**Explanation:  
Testcase1:** Largest number is 93100 which can be formed from array digits.

\*\* For More Input/Output Examples Use ['Expected Output'](https://practice.geeksforgeeks.org/problems/form-largest-number-from-digits/0#ExpectOP) option \*\*

[Author: harshitsidhwa](https://auth.geeksforgeeks.org/user/harshitsidhwa/practice/)

<https://practice.geeksforgeeks.org/problems/form-largest-number-from-digits/0>

#include <iostream>

#include <stdio.h>

using namespace std;

int main() {

int t;

scanf("%d", &t);

while(t--) {

int n;

scanf("%d", &n);

int hash[10];

for(int i =0; i<=9; i++) {

hash[i] = 0;

}

int a[n];

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

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

int elem;

scanf("%d", &elem);

while(elem > 0) {

hash[elem % 10]++;

elem /= 10;

}

}

string concat = "";

for(int i = 9; i>=0; i--){

for(int j =0; j < hash[i]; j++) {

concat += to\_string(i);

}

}

cout << concat << endl;

}

return 0;

}