**k largest elements**

[sorting](http://www.practice.geeksforgeeks.org/tag-page.php?tag=sorting&isCmp=0)[Amazon](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Amazon&isCmp=1)

Given an array, print k largest elements from the array.

**Input:**

The first line of input contains an integer T denoting the number of test cases.  
The first line of each test case is N and k, N is the size of array and K is the largest elements to be returned.  
The second line of each test case contains N input C[i].  
  
**Output:**

Print the k largest element in descending order.  
  
**Constraints:**

1 ≤ T ≤ 50  
1 ≤ N ≤ 100  
K ≤ N  
1 ≤ C[i] ≤ 1000  
  
**Example:**

**Input:**  
2  
5 2  
12 5 787 1 23  
7 3  
1 23 12 9 30 2 50

**Output:**  
787 23  
50 30 23

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

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=601>

#include <iostream>

#include <stdio.h>

#include <vector>

using namespace std;

int main() {

    int t;

    scanf("%d", &t);

    while(t--) {

      int n,k;

      scanf("%d %d", &n, &k);

      std::vector<int> v;

      for(int i=0; i <n; i++) {

         int elem;

         scanf("%d", &elem);

         v.push\_back(elem);

      }

      std::sort(v.begin(), v.end());

      std::reverse(v.begin(), v.end());

      for(int i =0; i < k; i++) {

         printf("%d ", v[i]);

        }

        printf("**\n**");

    }

 return 0;

}