[A - Better than everyone](https://vjudge.net/problem/HackerRank-better-than-everyone" \t "_blank)

Given N tasks each requiring Ai minutes, students need to complete at least K tasks. Obviously, students are choosing the K tasks which sum to the least time.

You want to be better than all of them at the same time not overload yourself with too much work or spend too much time.

Find the total minimum minutes that you will spend and is the least possible amount of time greater than others.

**Input Format**

First line contains T, the number of testcases.

For each testcase, first line contains N and K

Next line contains N space separated elements

**Constraints**

**1 <= T <= 500**

**2 <= N <= 50**

**1 <= Ai <=100**

**Output Format**

For each testcase print the required sum

**Sample Input 0**

2

2 1

1 2

6 3

77 44 47 74 7 4

**Sample Output 0**

2

58

HINT: Sort array in ascending order.  
i.e. a0, a1, a2, a3, …. ,ak-1, ak, ….., an-1`  
Now answer we are looking for is nothing but  
(first k-1 elements, skip kth element, include k+1th element)  
(a[0] + a[1] + …. + a[k-2]) + a[k]

#include <iostream>

#include <vector>

#include <algorithm>

using namespace *std*;

int main(void) {

*ios\_base*::*sync\_with\_stdio*(false);

*cin*.*tie*(nullptr);

*cout*.*tie*(nullptr);

int t; *cin* >> t;

while (t--) {

int n, k; *cin* >> n >> k;

*vector*<int> tasks(n);

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

*cin* >> tasks[i];

*sort*(tasks.*begin*(), tasks.*end*());

int sum1 = 0;

int sum2 = 0;

for (auto i = 0; i <= k; i++) {

sum1 += tasks[i];

}

sum2 = sum1 - tasks[k - 1];

*cout* << *min*(sum2, sum1) << "\n";

}

return 0;

}