Practical-5

AIM: Chef and his little brother are playing with sticks. They have total N sticks. Length of i-th stick is Ai. Chef asks his brother to choose any four sticks and to make a rectangle with those sticks its sides. Chef warns his brother to not to break any of the sticks, he has to use sticks as a whole. Also, he wants that the rectangle formed should have the maximum possible area among all the rectangles that Chef's brother can make.

Chef's little brother takes this challenge up and overcomes it. Can you also do so? That is, you have to tell whether it is even possible to create a rectangle? If yes, then you have to tell the maximum possible area of rectangle.

* Program

#include<bits/stdc++.h>

using namespace std;

int largestArea(vector<int> stickLength, int n){

int l = 0, b = 0;

sort(stickLength.begin(), stickLength.end(), greater<int>());

for(int i=0; i < n-1; i++){

if(stickLength[i] == stickLength[i+1]){

if(l < stickLength[i])

l = stickLength[i];

else if(b < stickLength[i])

b = stickLength[i];

i++;

}

}

if(l > 0 && b > 0)

return (l\*b);

else

return -1;

}

int main(){

cout << "This program is developed by 22CE097\_ShivangPatel" << endl;

int t, n;

cout << "Enter number of test cases : ";

cin >> t;

while(t>0){

cout << "Enter number of sticks : ";

cin >> n;

int a;

vector<int> stickLength;

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

cin >> a;

stickLength.push\_back(a);

}

cout << largestArea(stickLength, n) << endl;

t--;

}

return 0;

}

Output



* Conclusion

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_

Student Signature Faculty Signature Marks