

Courses V

Placement V

亷

Data Science V







Editorial

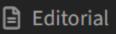
(Submissions

GATE **✓**











Subm

Compile & Run

Custom Input

Aggressive Cows □

Difficulty: Medium

Accuracy: **59.57**%

Submissions: 169K+ Points: 4

Average Time: 30m

You are given an array with unique elements of stalls[], which denote the positions of **stalls**. You are also given an integer **k** which denotes the number of aggressive cows. The task is to assign stalls to k cows such that the minimum distance between any two of them is the maximum possible.

Examples:

Input: stalls[] = [1, 2, 4, 8, 9], k = 3

Output: 3

Explanation: The first cow can be placed at stalls[0],

the second cow can be placed at stalls[2] and

the third cow can be placed at stalls[3].

The minimum distance between cows in this case is 3, which is the largest among all possible ways.

🌣 Start Timer 🕞 C++(12)bool yes(vector<int> &stalls, int k, int dis) 3 4 int c=1,prev=stalls[0]; 5 for(int i=1;i<stalls.size();i++)</pre> 6 7 if(stalls[i]-prev>=dis) 8 9 c++; 10 prev=stalls[i]; 11 12 13 return c>=k; 14 15 int aggressiveCows(vector<int> &stalls, int k) { 16 17 // code here 18 sort(stalls.begin(),stalls.end()); int n= stalls.size(); 19 int l=1,h=stalls[n-1]-stalls[0]; 20 21 int ans=0; while(1<=h) 22 23 int mid=1+(h-1)/2; 24 if(yes(stalls,k,mid)) 25 26 ans=mid; 27 l=mid+1; 28 29 else{h=mid-1:} 30

Input: stalls[] = [10, 1, 2, 7, 5], k = 3

Input: $arr[] = [12 \ 34 \ 67 \ 90] \ k = 2$