Aggressive Bull Problem 1

locked

Problem Submissions Leaderboard Discussions

Anand has built a new barn, with N (2 <= N <= 100,000) stalls. The stalls are located along a straight line at positions a1,...,aN (0 <= ai <= 1,000,000). His X (2 <= X <= N) bulls don't like this barn layout and become aggressive towards each other once put into a stall. To prevent the bull from hurting each other, Anand wants to assign the bulls to the stalls, such that the minimum distance between any two of them is as large as possible. What is the largest minimum distance

Input Format

- t – the number of test cases, then t test cases follows.
- Line 1: Two space-separated integers: N and C
- Lines 2..N+1: Line i+1 contains an integer stall location, ai

Constraints

(2 <= N <= 100,000)
(0 <= xi <= 1,000,000,000

Output Format

For each test case output one integer: the largest minimum distance.

Sample Input 0

```
1
5 3
1
2
8
4
9
```

Sample Output 0

```
3
```

Submissions: 1
Max Score: 20
Difficulty: Medium

Rate This Challenge:

More

Current Buffer (saved locally, editable)

C++

```
#include <bits/stdc++.h>
using namespace std;
```

```
3
4  int N,C;
5
6  int check(int num,int stalls[])
7  {
8      int cows=1,pos=stalls[0];
9      for (int i=1; i<N; i++)
10     {
11         if (stalls[i]-pos>=num)
12         {
13             pos=stalls[i];
14             cows++;
15             if (cows==C)
16                 return 1;
17         }
18     }
19     return 0;
20 }
21
22 int binarySearch(int stalls[])
23 {
24     int start=0,end=stalls[N-1],max=-1;
25     while (end>start)
26     {
27         int mid=(start+end)/2;
28         if (check(mid,stalls)==1)
29         {
30             if (mid>max)
31                 max=mid;
32             start=mid+1;
33         }
34         else
35             end=mid;
36     }
37     return max;
38 }
39
40 int main()
41 {
42     int t;
43     cin>>t;
44
45     while (t--)
46     {
47         cin>>N>>C;
48
```

```
        int stalls[N];

        for (int i=0; i<N; i++)
            cin>>stalls[i];

        sort(stalls,stalls+N);

        int k=binarySearch(stalls);

        cout<<k;
    }
    return 0;
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Testcase 0 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
1
5 3
1
2
8
4
9
```

Your Output (stdout)

3

Expected Output

3