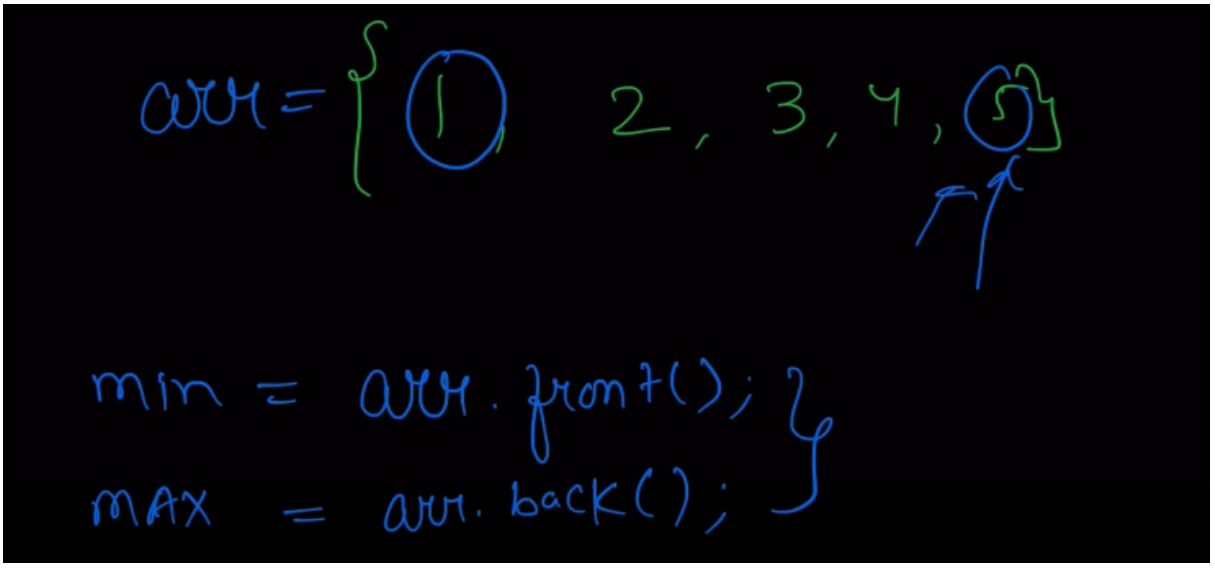
624. Maximum Distance in Arrays - 16/08/24 (medium)

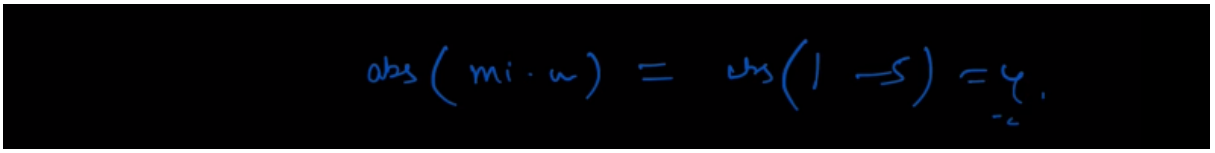
greedy approach

You are given `m` `arrays`, where each array is sorted in **ascending order**.

You can pick up two integers from two different arrays (each array picks one) and calculate the distance. We define the distance between two integers `a` and `b` to be their absolute difference `|a - b|`.

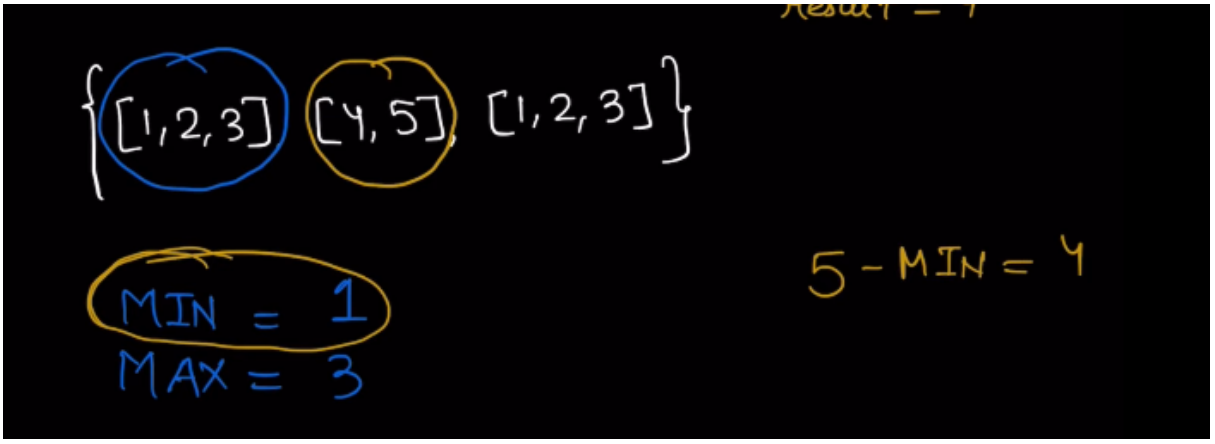
Return *the maximum distance*.



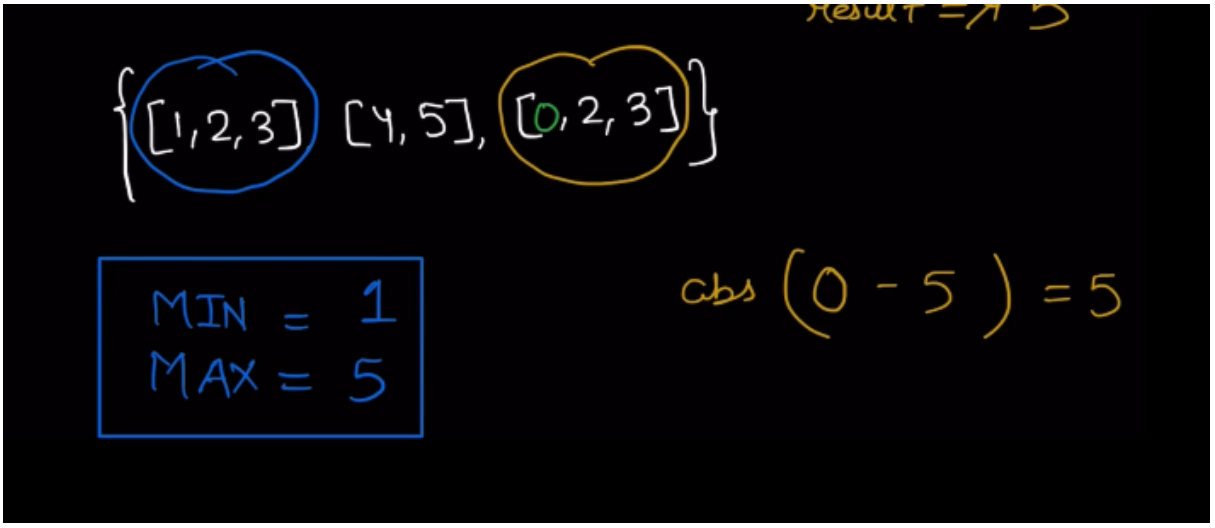


it is easy for sinngle array now to find multiple array

we use this technique



now we find by
current min and past maximum to find the answer



//OPTIONAL SOLTION

we cna use brute force to solve it
curent array ka min dusre array ke max se minus karo
ya curerrent element ke maxium ka dusre element ke minimum se comapre kar ke answer leleo

now code it

Solution

```
class Solution {
public:
    int maxDistance(vector<vector<int>>& arrays) {
        int globalMin = arrays[0].front(); // Initialize to t
        int globalMax = arrays[0].back();  // Initialize to t
        int result = 0;

        for (int i = 1; i < arrays.size(); ++i) {
            int currMin = arrays[i].front();
            int currMax = arrays[i].back();

            // Calculate the maximum distance using the globa
            result = max({result, abs(currMax - globalMin), a

            // Update global min and max
            globalMin = min(globalMin, currMin);
            globalMax = max(globalMax, currMax);
        }

        return result;
    }
};
```