

Classwork

Search In Rotated Sorted Array [Coding Ninjas]

Question Link:-

https://www.codingninjas.com/studio/problems/search-in-rotated-sorted-array_630450?leftPanelTab=0

The screenshot displays the Coding Ninjas studio interface. On the left, the 'Current Submission' panel shows a 'Correct Answer' status with 7/7 test cases passed, a score of 80/80, and a runtime of 2426 ms. Below this is a runtime graph showing performance relative to other solutions. The main editor on the right contains a Java solution for the problem. The solution uses a binary search approach to find the minimum element in the rotated array and then searches for the target key. The interface includes tabs for Topics, Problem, Submissions, Solution, and Discuss, along with a console on the right.

```
1 public class Solution {
2     public static int search(int arr[], int key) {
3         // Write your code here.
4         int s=0,e=arr.length-1;
5         while(s<=e)
6         {
7             int mid=s+(e-s)/2;
8             if (arr[mid] == key)
9             {
10                 return mid;
11             }
12             if (arr[s] <= arr[mid])
13             {
14                 if (arr[s] <= key && key <= arr[mid])
15                 {
16                     e = mid - 1;
17                 }
18                 else
19                 {
20                     s = mid + 1;
21                 }
22             }
23         }
24     }
25 }
```

Java Code

```
public class Solution {
    public static int search(int arr[], int key) {
        // Write your code here.
        int s=0,e=arr.length-1;
        while(s<=e)
        {
            int mid=s+(e-s)/2;
            if (arr[mid] == key)
            {
                return mid;
            }
            if (arr[s] <= arr[mid])
            {
                if (arr[s] <= key && key <= arr[mid])
                {
                    e = mid - 1;
                }
                else
                {
                    s = mid + 1;
                }
            }
        }
    }
}
```

```

    }
}

else if(arr[s] >= arr[mid])
{

    if (arr[mid] <= key && key <= arr[e])
    {
        s = mid + 1 ;
    }
    else
    {
        e = mid - 1;
    }
}

return -1;
}
}
}

```

Sqrt(x) [LeetCode]

Question link:-

<https://leetcode.com/problems/sqrtx/description/>

69. Sqrt(x) Solved

Easy Topics Companies Hint

Given a non-negative integer x , return the square root of x rounded down to the nearest integer. The returned integer should be **non-negative** as well.

You **must not use** any built-in exponent function or operator.

7.4K 4.3K ☆ ↺ ⌂

Testcase Result X

Accepted

Runtime 1 ms
Beats 97.45% of users with Java

Memory 39.36 MB
Beats 82.14% of users with Java

More challenges

- 367. Valid Perfect Square

Code

```

1 class Solution {
2     public int mySqrt(int x) {
3         if(x==0){
4             return 0;
5         }
6
7         int start=1;
8         int end=x;
9         int answer=-1;
10        int mid=start+(end-start)/2;
11        while(start<end){
12            if(mid==x/mid){
13                return mid;
14            }else if(mid>x/mid){
15                end=mid-1;
16            }else{
17                answer=mid;
18                start=mid+1;
19            }
20            mid=start+(end-start)/2;
21        }
22        return answer;
23    }
24 }

```

Saved to local Ln 20, Col 37

Java Code

```
class Solution {
    public int mySqrt(int x) {
        if(x==0){
            return 0;
        }

        int start=1;
        int end=x;
        int answer=-1;
        int mid=start+(end-start)/2;
        while(start<=end){
            if(mid==x/mid){
                return mid;
            }else if(mid>x/mid){
                end=mid-1;
            }else{
                answer=mid;
                start=mid+1;
            }
            mid=start+(end-start)/2;
        }
        return answer;
    }
}
```