```
int findpivot(vector<int>& arr){
    int s=0;
    int e=arr.size()-1;
    int mid=(s+e)/2;
    while(s<e){
        if(arr[mid]>=arr[0]){
            s=mid+1;
            }
        else{
            e=mid;
            }
            mid=(s+e)/2;
        }
        return s;
    }
int binarysearch(vector<int>& arr,int s,int e,int k){
    int mid=(s+e)/2;
    while(s<=e){
        if(arr[mid]==k){
            return mid;
            }
        if(k<arr[mid]){
            e=mid-1;
            }
        else{

            s=mid+1;

            }

            mid=(s+e)/2;

        }

        return -1;

    }
int search(vector<int>& arr, int n, int k)
{
    int pivot=findpivot(arr);
    if(k>=arr[pivot] && k<= arr[n-1]){
        return binarysearch(arr,pivot,n-1,k);
    }
    else{
        return binarysearch(arr,0,pivot-1,k);
    }
}
```