2.5 KTH MISSING POSITIVE NUMBER

Question:

Given an array arr of positive integers sorted in a strictly increasing order, and an integer k. return the kth positive integer that is missing from this array.

AIM

To find the kth missing positive integer that does not appear in a strictly increasing array arr.

ALGORITHM

1. Initialize a counter missing = 0 to count missing numbers.
2. Traverse natural numbers starting from 1.
   - If the current number exists in arr, skip it.
   - Otherwise, increase missing.
3. Stop when missing == k.
4. Return the current number.

PROGRAM

```
def find_kth_missing(arr, k):
    missing = 0
    current = 1
    idx = 0
    while True:
        if idx < len(arr) and arr[idx] == current:
            idx += 1
        else:
            missing += 1
            if missing == k:
                return current
        current += 1

def run_kth_missing():
    arr = list(map(int, input("Enter sorted array: ").split()))
    k = int(input("Enter k: "))
    print("K-th missing positive number:", find_kth_missing(arr, k))

run_kth_missing()
```

Input:

arr = [2,3,4,7,11]

k = 5

Output:

```
Enter sorted array: 2 3 4 7 11
Enter k: 3
K-th missing positive number: 6
>>>
```

RESULT:

Thus the program is successfully executed and the output is verified.

PERFORMANCE ANALYSIS:

- Time Complexity: O(log n) (binary search)
- Space Complexity: O(1)