Find element at a given Index

Given an array **arr** of integers and an index **key**(0-based index). Your task is to return the element present at the index key in the array.

Examples:

Input: key = 2 , arr = [10, 20, 30, 40, 50]
Output: 30
Explanation: The value of arr[2] is 30 .

Expected Time Complexity: O(1)
Expected Auxiliary Space: O(1)

Constraints:

0 ≤ key ≤ arr.size - 1

1 ≤ arr.size ≤ 10^6

1 ≤ arr[i] ≤ 10^9

For Input:
5
1 1 2 2 3 4 5 5 6 7

Your Output:

```
#include <iostream>
#include <vector>
using namespace std;

class Solution
{
public:
    int findElementAtIndex(int key, vector<int> &arr)
    {
        for (int i = 0; i < arr.size(); ++i)
        {
            if (arr[i] == key) return i;
        }
        return -1;
    }
};

int main()
{
    vector<int> arr = {1, 1, 2, 2, 3, 4, 5, 5, 6, 7};
    Solution solution;

    int key = 5;
    int index = solution.findElementAtIndex(key, arr);

    if (index != -1)
        cout << "Element " << key << " found at index " << index << endl;
    else
        cout << "Element not found" << endl;

    return 0;
}
```