Peak element in C++

```
#include <iostream>
#include <vector>

using namespace std;

int findPeakElement(const vector<int>& arr) {
    int low = 0, high = arr.size() - 1;

    while (low <= high) {
        int mid = (low + high) / 2;

        if ((mid == 0 || arr[mid - 1] <= arr[mid])
            && (mid == arr.size() - 1 || arr[mid + 1] <= arr[mid])) {
            return mid;
        }

        if (mid > 0 && arr[mid - 1] >= arr[mid]) {
            high = mid - 1;
        } else {
            low = mid + 1;
        }
    }

    return -1; // Peak element not found
}

int main() {
    vector<int> arr = {10, 7, 8, 20, 12};
    cout << findPeakElement(arr) << endl;
    return 0;
}
```

Dry Run Table:

Iteration	low	high	mid	arr[mid-1]	arr[mid]	arr[mid+1]	Condition Met	Action
1	0	4	2	7	8	20	Right neighbor > mid	low = mid + 1 = 3
2	3	4	3	8	20	12	Both neighbors ≤ mid → **peak found!**	Return 3

✅ **Output:**

3

3