Count Zeroes In Sorted Matrix in C++

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

class CountZerosInASortedMatrix {
public:
    static int countZeros(vector<vector<int>>& mat) {
        int n = mat.size();
        int i = 0;
        int j = n - 1;
        int countZeros = 0;

        while (i < n && j >= 0) {
            if (mat[i][j] == 1) {
                j--;
            } else {
                countZeros += j + 1;
                i++;
            }
        }

        return countZeros;
    }
};

int main() {
    // Hardcoded input
    int n = 5;
    vector<vector<int>> mat = {
        {0, 0, 0, 1, 1},
        {0, 0, 0, 1, 1},
        {0, 0, 1, 1, 1},
        {0, 1, 1, 1, 1},
        {0, 1, 1, 1, 1}
    };

    // Call the countZeros method to count zeros
    int result =
CountZerosInASortedMatrix::countZeros(mat);

    // Print the result
    cout << "Number of zeros in the sorted matrix: " <<
result << endl;

    return 0;
}
```

Dry Run Table

Matrix:

0 0 0 1 1
0 0 0 1 1
0 0 1 1 1
0 1 1 1 1
0 1 1 1 1

i	j	mat[i][j]	Action	Zeros Count
0	4	1	j-- → 3	0
0	3	1	j-- → 2	0
0	2	0	count += 2+1=3, i++	3
1	2	0	count += 2+1=3, i++	6
2	2	1	j-- → 1	6
2	1	0	count += 1+1=2, i++	8
3	1	1	j-- → 0	8
3	0	0	count += 0+1=1, i++	9
4	0	0	count += 0+1=1, i++	10

✅ **Final Output:**

Number of zeros in the sorted matrix: 10

Number of zeros in the sorted matrix: 10