|  |  |
| --- | --- |
|  | **DEPARTMENT OF COMPUTER ENGINEERING** |

**Experiment No. 05**

|  |  |
| --- | --- |
| Semester | B.E. Semester VII – Computer Engineering |
| Subject | Big Data Analysis |
| Subject Professor In-charge | Prof. Pankaj Vanvari |
| Lab Professor In-charge | Dr. Umesh Kulkarni |
| Academic Year | 2024-25 |

|  |  |
| --- | --- |
| Student Name | Deep Salunkhe |
| Roll Number | 21102A0014 |

**Title:** Flajolet Martin

#include <iostream>

#include <vector>

#include <cmath>

#include <climits>

#include <algorithm>

*// Simple hash function*

int basicHash(int x) {

    return (x \* 2654435761) % INT\_MAX; *// A basic multiplicative hash*

}

*// Function to count the number of trailing zeros in binary representation*

int countTrailingZeros(int n) {

    int count = 0;

    while (n > 0) {

        if (n & 1) {

            break;

        }

        count++;

        n >>= 1;

    }

    return count;

}

*// Flajolet-Martin algorithm*

int flajoletMartin(const std::vector<int>& data) {

    int maxZeros = 0;

    for (int x : data) {

        int hashedValue = basicHash(x);

        int trailingZeros = countTrailingZeros(hashedValue);

        maxZeros = std::max(maxZeros, trailingZeros);

    }

    return std::pow(2, maxZeros); *// Estimate the number of distinct elements*

}

int main() {

    int n;

    std::cout << "Enter the number of elements in the vector: ";

    std::cin >> n;

    std::vector<int> data(n);

    std::cout << "Enter the elements of the vector: ";

    for (int i = 0; i < n; ++i) {

        std::cin >> data[i];

    }

*// Estimate the number of distinct elements using Flajolet-Martin*

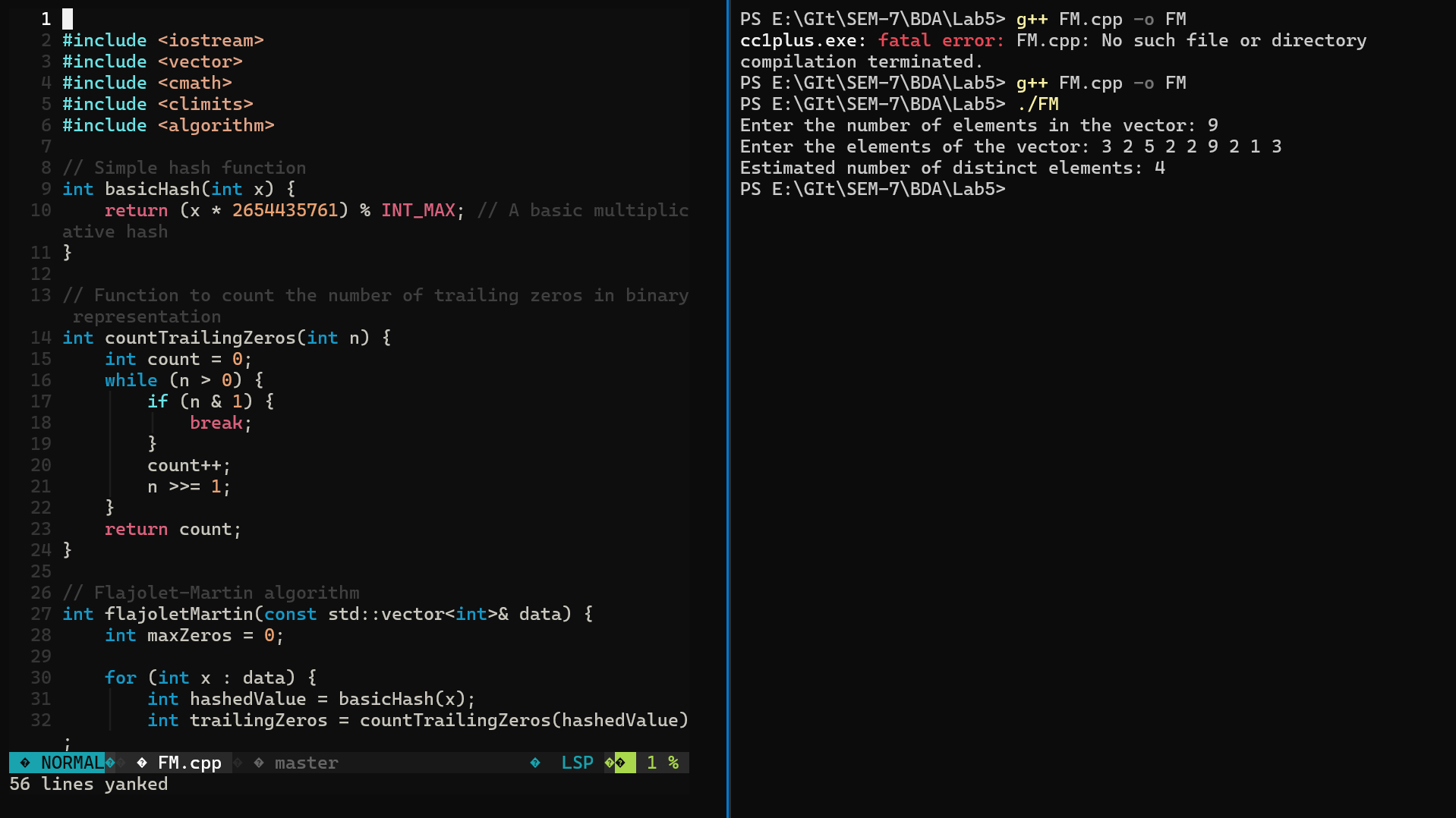
    int distinctCountEstimate = flajoletMartin(data);

    std::cout << "Estimated number of distinct elements: " << distinctCountEstimate << std::endl;

    return 0;

}

**Output:**

****