# AIR UNIVERSITY
## DEPARTMENT OF MECHATRONICS ENGINEERING
## SEMESTER: SPRING 2025
## Computer Programming CE-112

## Assignment-3

**Instructor: Umer Farooq** **Due Date:** 11th May 2025
Soft Copy Time:10:00 PM

**Cover Page must be filled as per attached.**
**Name:** Syeda Arzoo Zahra **Name in ASCII( HEX):** 41 72 7A 6F 6F
**Rollno: 242125** **Rollno:in 4bit Hex:** 3B16D

| Question No: | Write Status(Complete/Not complete and issue that you faced (compulsory Question will be checked on bases of this index) | Remarks Must be filled for every question | |
|---|---|---|---|
| 1 | Completed | Difficult | |
| 2 | Completed | Difficult | |
| 3 | Completed | Difficult | |
| 4 | Completed | Difficult | |
| 5 | Completed | Difficult | |
| 6 | Completed | Difficult | |


I hear and I forget. I see and I remember. I do and I understand.

Confucius

# Functions and Array

1. Write and test the following function that returns through its reference parameters both the maximum and the minimum values stored in an array: void get Extremes(float& min, float& max, float a[], int n);

## Solution:

cp A3 Q1.cpp

```cpp
#include <iostream>
using namespace std;
//ARZOO
 void getExtremes(float& min, float& max, float a[], int n) {
    if (n <= 0) {
        cout << "Array size must be greater than 0." << endl;
        return;
    }

    min = max = a[0];

    for (int i = 1; i < n; ++i) {
        if (a[i] < min)
            min = a[i];
        if (a[i] > max)
            max = a[i];
    }
}

int main() {
    float arr[] = {22.2, 6.1, 89.8, 12.3, 0.9, -0.5};
    int n = sizeof(arr) / sizeof(arr[0]);
    float minVal, maxVal;

    getExtremes(minVal, maxVal, arr, n);

    cout << "Minimum value: " << minVal << endl;
    cout << "Maximum value: " << maxVal << endl;

    return 0;
}
```

## Result:

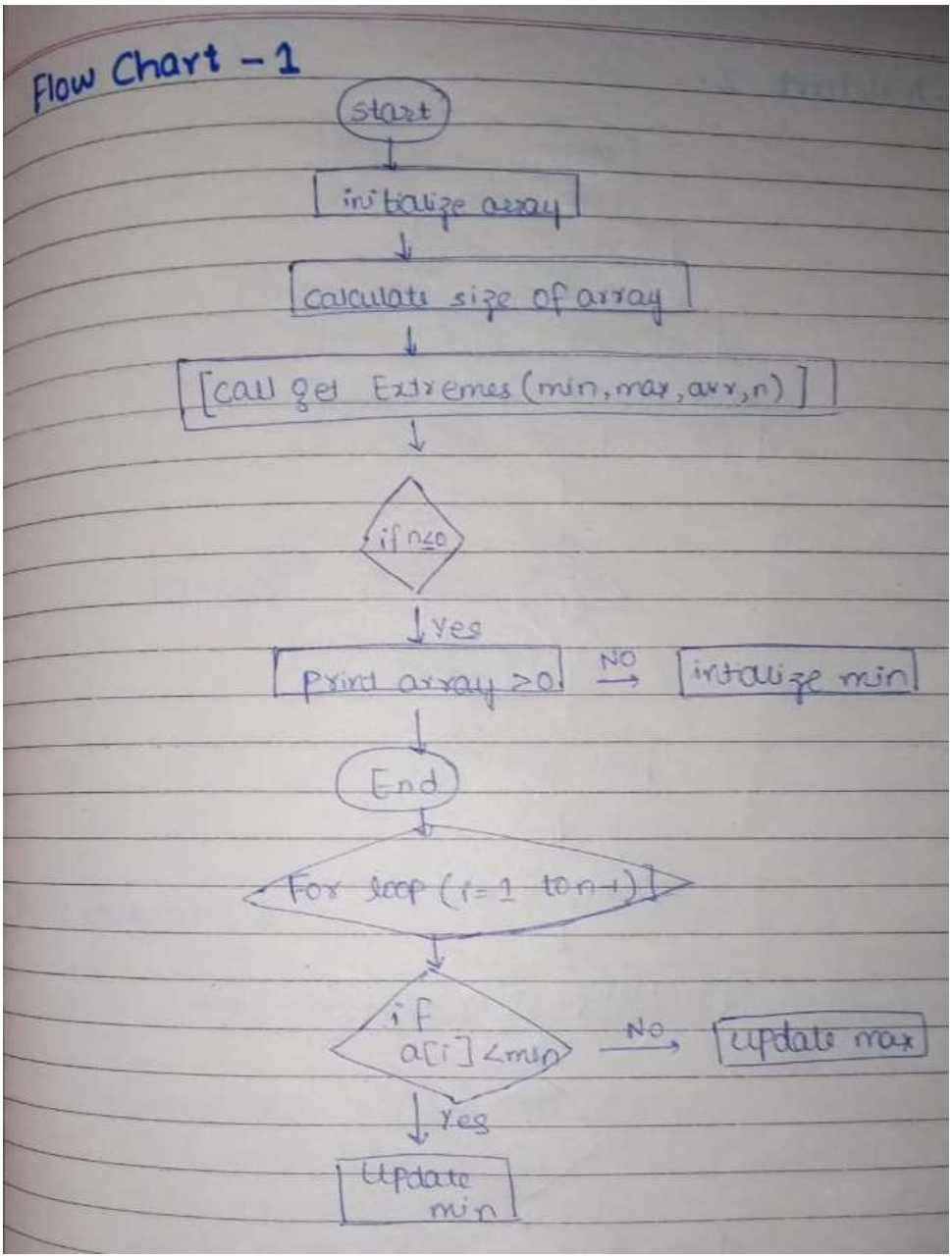


I hear and I forget. I see and I remember. I do and I understand.

Confucius

```
C:\Users\DELL\OneDrive\Documentos\cp A3 Q1.exe
Minimum value: -0.5
Maximum value: 89.8

--------------------------------
Process exited after 0.1842 seconds with return value 0
Press any key to continue . . .
```

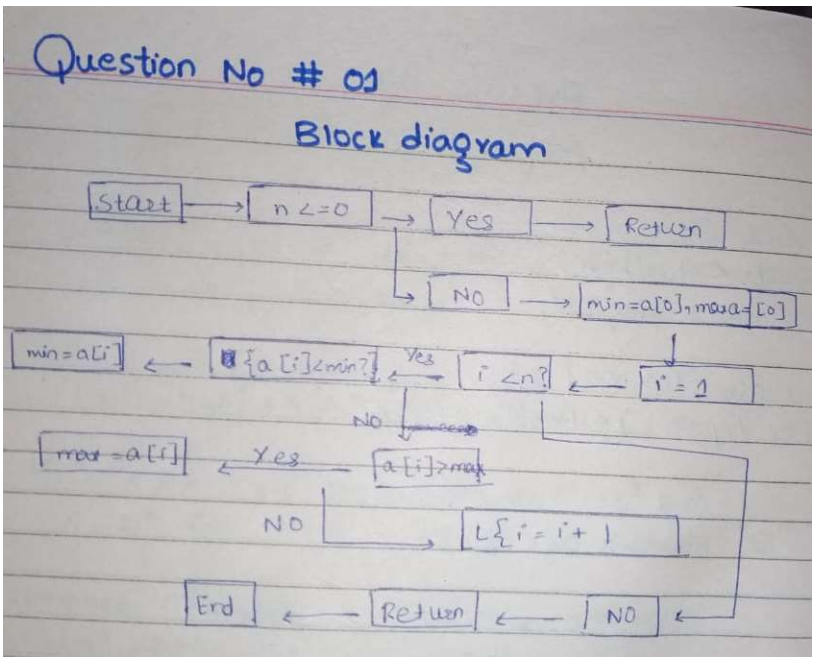
## Flowchart:




 I hear and I forget. I see and I remember. I do and I understand.

Confucius

## Block diagram:




2. Write and test the following function: double stdev(double x[], int n); The function

$$s = \sqrt{\frac{\sum_{i=0}^{n-1}(x_i - \bar{x})^2}{n-1}}$$

returns the standard deviation of a data set of n numbers $x_0, \ldots, x_{n-1}$

## Solution:

main.cpp

Share Run

```cpp
#include <iostream>
#include <vector>
#include <math.h>
#include <numeric>
using namespace std;
//arzoo
double stdev(const double x[], int n) {
    if (n <= 1) return 0;
    double sum = accumulate(x, x + n, 0.0);
    double mean = sum / n;
    double sq_diff_sum = 0;
    for (int i = 0; i < n; ++i) sq_diff_sum += pow(x[i] - mean, 2);
    return sqrt(sq_diff_sum / (n - 1));
}
int main() {
    double data[] = {2, 4, 5, 4, 5, 6, 7, 8};
    int n = sizeof(data) / sizeof(data[0]);
    cout << "Standard Deviation: " << stdev(data, n) << endl;
    return 0;
}
```

**Result:**

Output

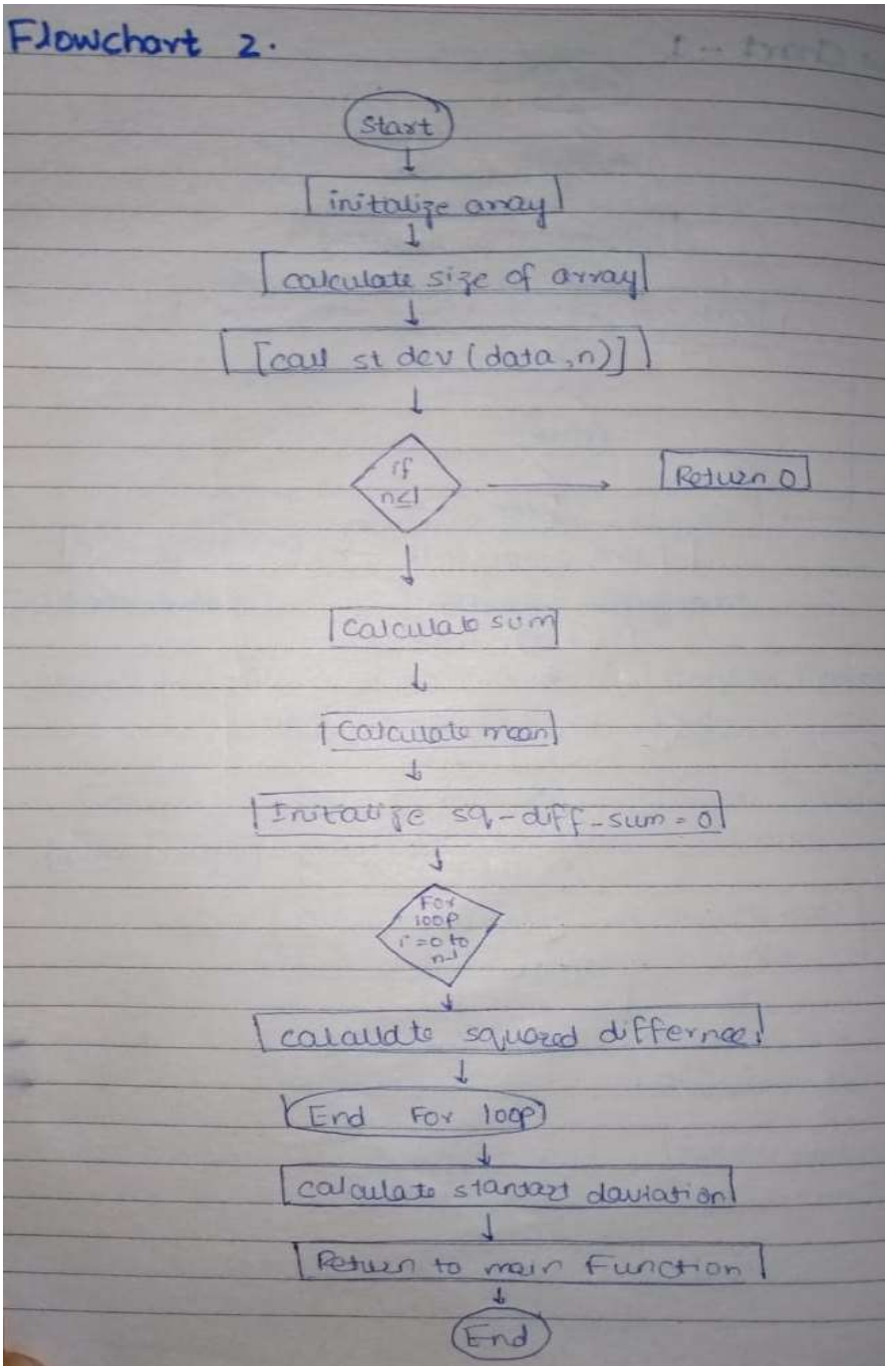
```
Standard Deviation: 1.88509


=== Code Execution Successful ===
```

**Flowchart:**



I hear and I forget. I see and I remember. I do and I understand.

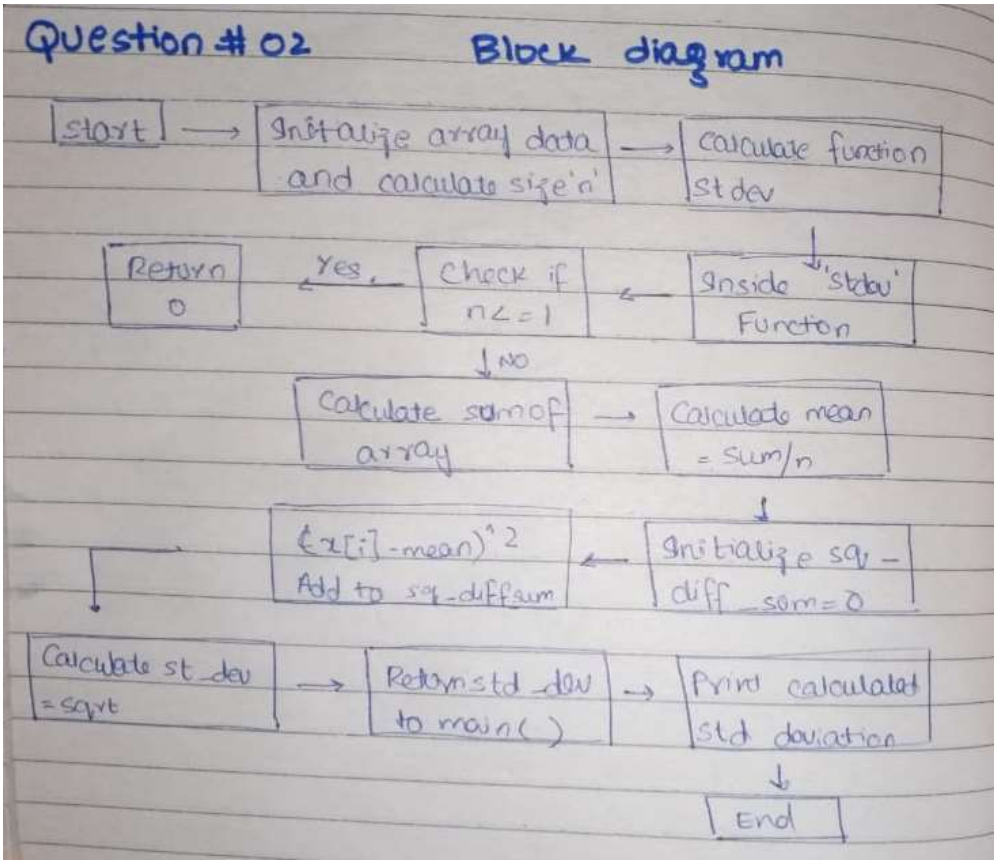
Confucius




## Block diagram:

 I hear and I forget. I see and I remember. I do and I understand.

Confucius




3. Write a and test a program to calculate ;write a program to multiply two 3x3 matrices.

## Solution:

main.cpp

Share Run

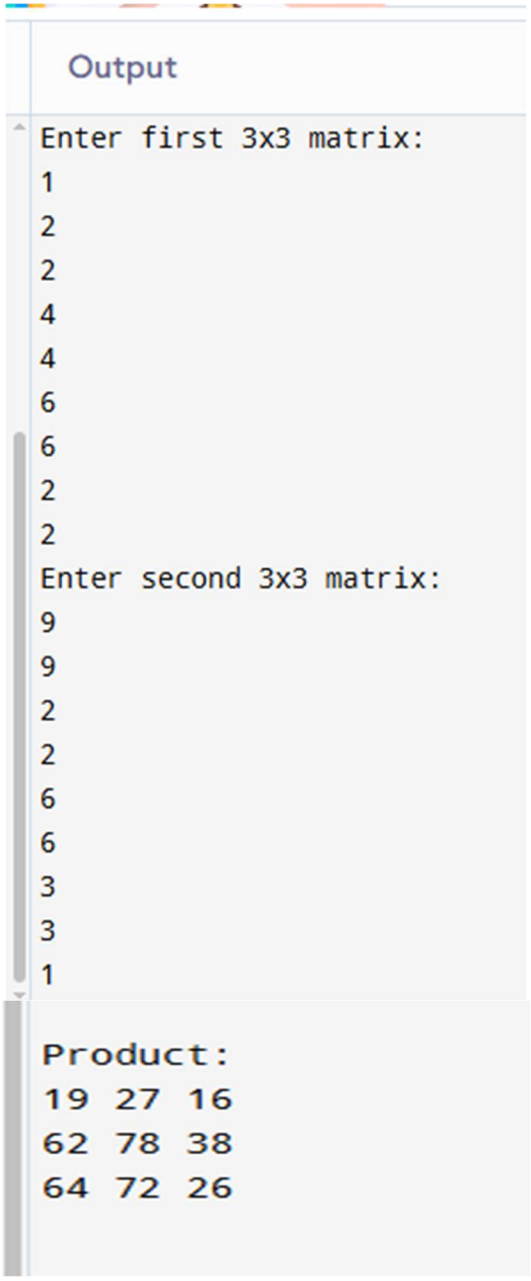
```cpp
#include <iostream>
#include <vector>
using namespace std;
//arzoo
using Mat3 = vector<std::vector<int>>;
Mat3 multiply(const Mat3& a, const Mat3& b) {
    Mat3 res(3,vector<int>(3, 0));
    for (int i = 0; i < 3; ++i)
        for (int j = 0; j < 3; ++j)
            for (int k = 0; k < 3; ++k)
                res[i][j] += a[i][k] * b[k][j];
    return res;
}
void print(const Mat3& m) {
    for (const auto& row : m) {
        for (int val : row) cout << val << " ";
        cout << endl;
    }
}

int main() {
    Mat3 mat1(3, vector<int>(3));
    Mat3 mat2(3, vector<int>(3));
   cout << "Enter first 3x3 matrix:\n";
    for (auto& row : mat1) for (int& val : row) cin >> val;
    cout << "Enter second 3x3 matrix:\n";
    for (auto& row : mat2) for (int& val : row) cin >> val;
    cout << "\nProduct:\n";
    print(multiply(mat1, mat2));
 return 0;
}
```

**Result:**



I hear and I forget. I see and I remember. I do and I understand.

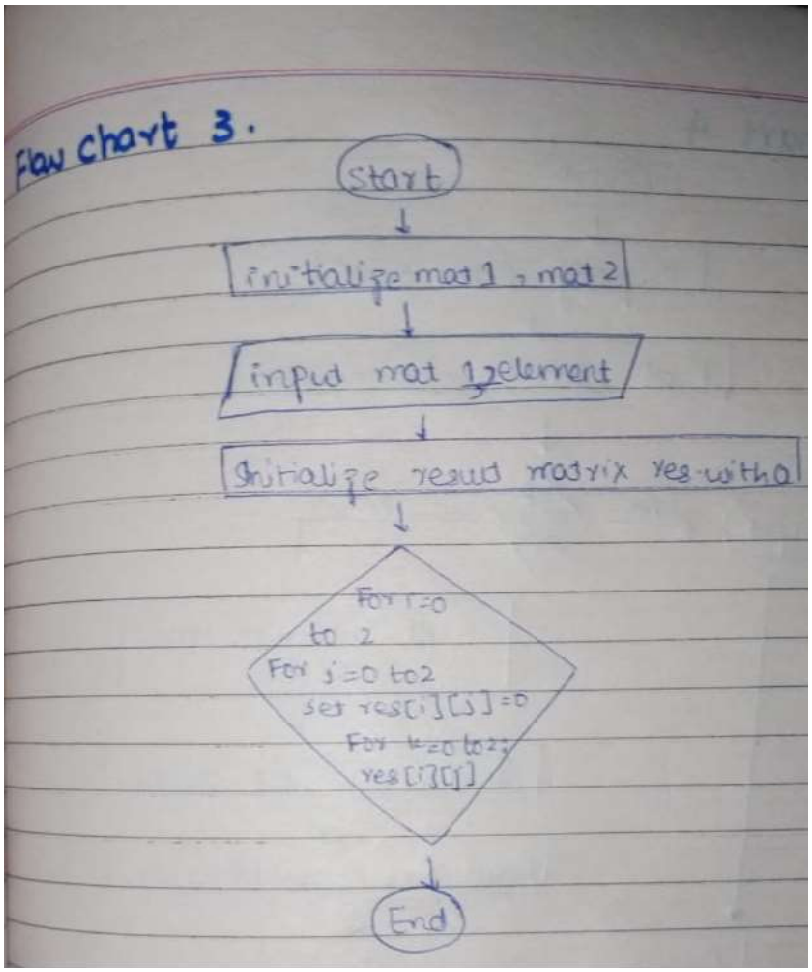
Confucius



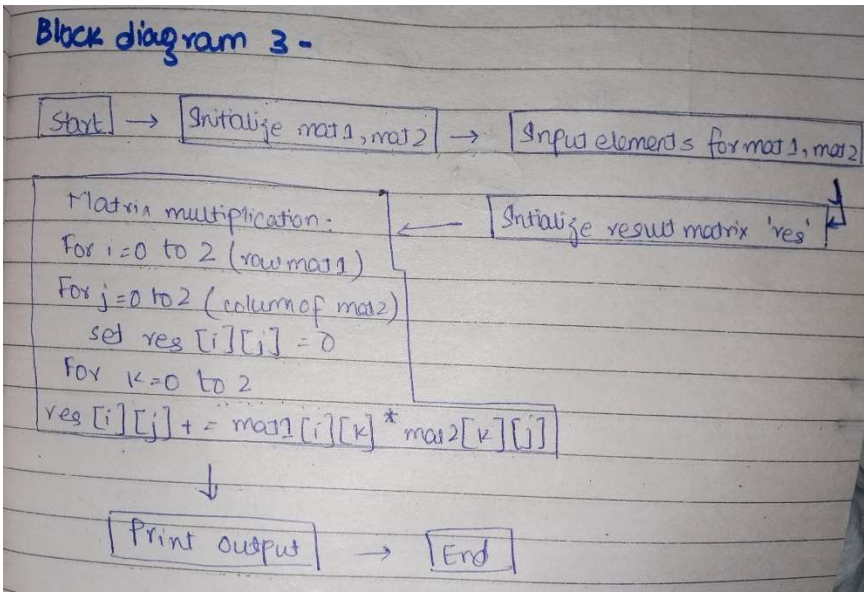

**Flowchart:**

 I hear and I forget. I see and I remember. I do and I understand.

Confucius




## Block diagram:




 I hear and I forget. I see and I remember. I do and I understand.

Confucius

4. Write a program that takes a string from the user. The program then calculates the
   - ✓ Total number of Characters
   - ✓ Total number of Spaces
   - ✓ Total number of Tabs
   - ✓ Total number of Lines

**Solution:**


```cpp
#include <iostream>
#include <string>
using namespace std;
//arzoo
int main() {
    string inputString;
    int charCount = 0;
    int spaceCount = 0;
    int tabCount = 0;
    int lineCount = 1;

    cout << "Enter a string: ";
    getline(cin, inputString);

    for (char c : inputString) {
        charCount++;
        if (c == ' ') {
            spaceCount++;
        } else if (c == '\t') {
            tabCount++;
```



I hear and I forget. I see and I remember. I do and I understand.

Confucius

```
21        } else if (c == '\n') {
22            lineCount++;
23        }
24    }
25    cout << "\n--- String Analysis ---" << endl;
26    cout << "Total number of Characters: " << charCount <<endl;
27    cout << "Total number of Spaces:" << spaceCount << endl;
28    cout << "Total number of Tabs:" << tabCount << endl;
29    cout << "Total number of Lines:" << lineCount << endl;
30    return 0;
31 }
```

**Result:**

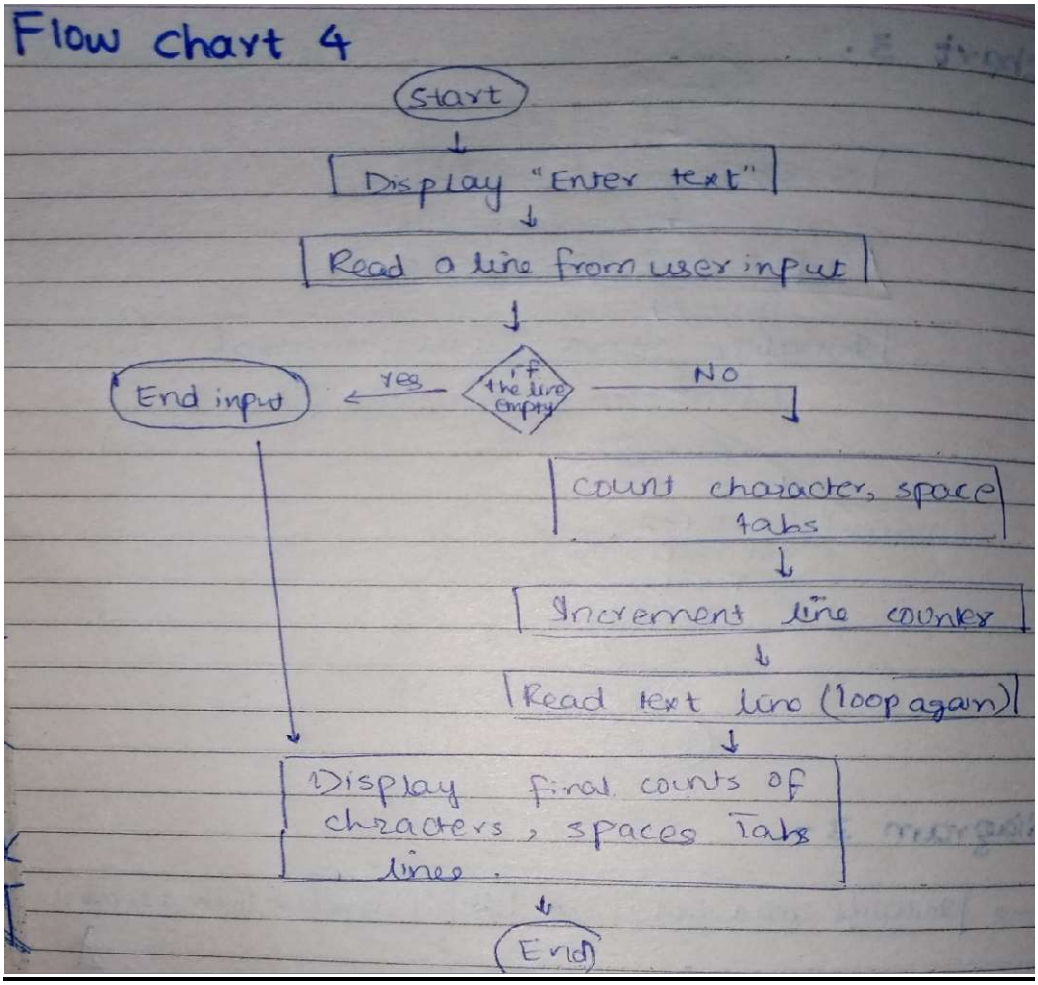
```
Output

Enter a string: arzoo

--- String Analysis ---
Total number of Characters: 5
Total number of Spaces:0
Total number of Tabs:0
Total number of Lines:1
```

**Flowchart:**

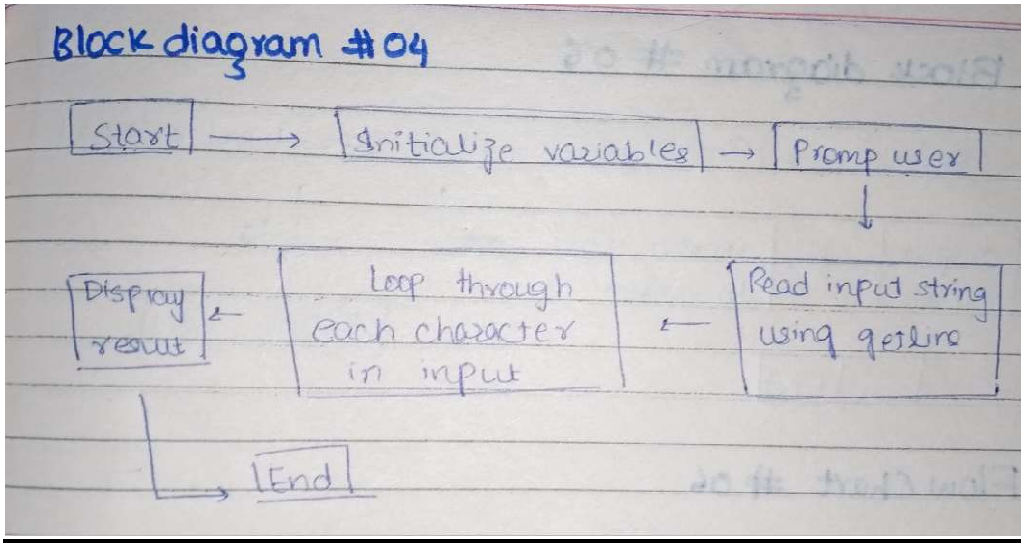


I hear and I forget. I see and I remember. I do and I understand.

Confucius




**Block Diagram:**




 I hear and I forget. I see and I remember. I do and I understand.

Confucius

5. Write a program to calculate **mean**, **median** and **mode** of an array.

## Solution:

main.cpp

Share Run

```cpp
#include <iostream>
#include <vector>
#include <algorithm>
#include <map>
//ARZOO
using namespace std;

double calculateMean(const vector<int>& arr) {
    double sum = 0;
    for (int num : arr) {
        sum += num;
    }
    return sum / arr.size();
}
double calculateMedian(vector<int> arr) {
    sort(arr.begin(), arr.end());
    int n = arr.size();
    if (n % 2 == 0) {
        return (arr[n / 2 - 1] + arr[n / 2]) / 2.0;
    } else {
```



I hear and I forget. I see and I remember. I do and I understand.

Confucius

main.cpp

Share Run

```
        return arr[n / 2];
    }
}
vector<int> calculateMode(const vector<int>& arr) {
    map<int, int> frequency;
    for (int num : arr) {
        frequency[num]++;
    }
    int maxCount = 0;
    for (const auto& pair : frequency) {
        if (pair.second > maxCount) {
            maxCount = pair.second;
        }
    }
    vector<int> modes;
    for (const auto& pair : frequency) {
        if (pair.second == maxCount) {
            modes.push_back(pair.first);
        }
    }
```

main.cpp

Share Run

```
    return modes;
}
int main() {
    vector<int> arr;
    int n, input;
    cout << "Enter the number of elements in the array: ";
    cin >> n;
    cout << "Enter the elements of the array: ";
    for (int i = 0; i < n; i++) {
        cin >> input;
        arr.push_back(input);
    }
    double mean = calculateMean(arr);
    double median = calculateMedian(arr);
    vector<int> mode = calculateMode(arr);

    cout << "Mean: " << mean << endl;
    cout << "Median: " << median << endl;
    cout << "Mode: ";
    for (int m : mode) {
```

 I hear and I forget. I see and I remember. I do and I understand.

Confucius

```
61          cout << m << " ";
62      }
63      cout << endl;
64      return 0;
65  }
```

**Result:**

Output

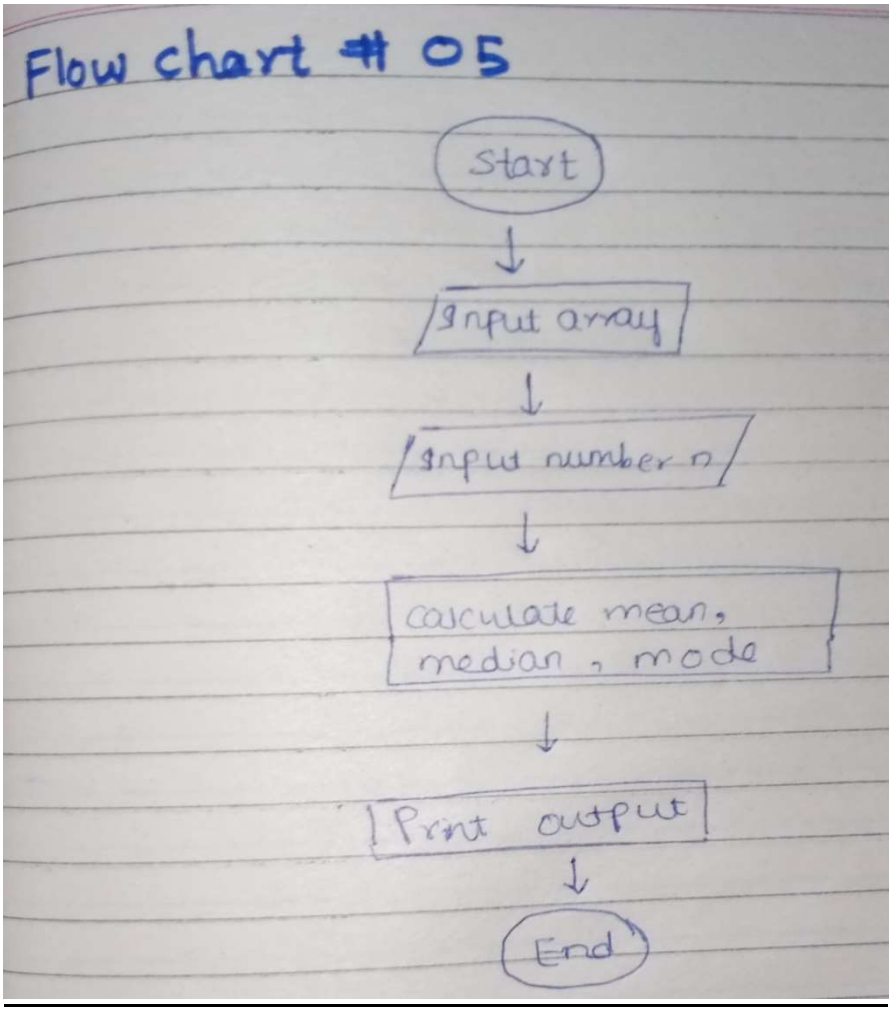
```
Enter the number of elements in the array: 8
Enter the elements of the array: 2
4
6
8
2

5
7
8
Mean: 5.25
Median: 5.5
Mode: 2 8
```

**Flowchart:**

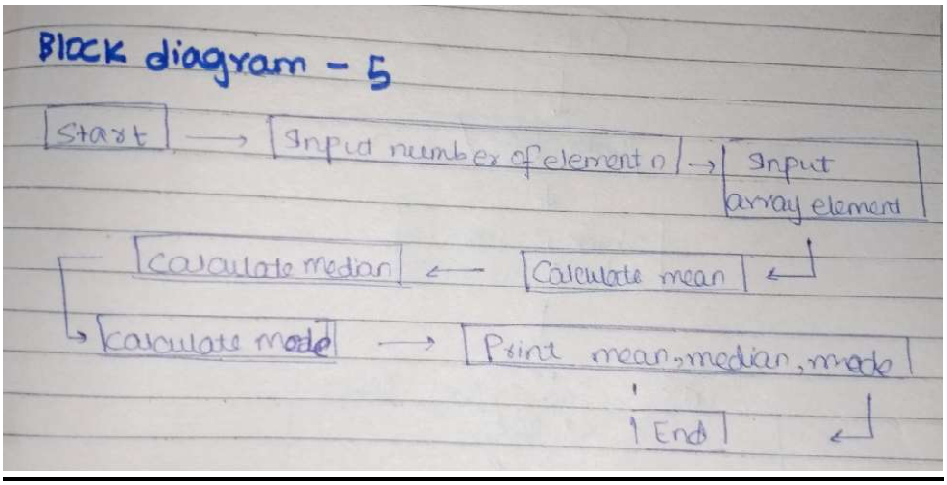


I hear and I forget. I see and I remember. I do and I understand.

Confucius




**Block diagram:**



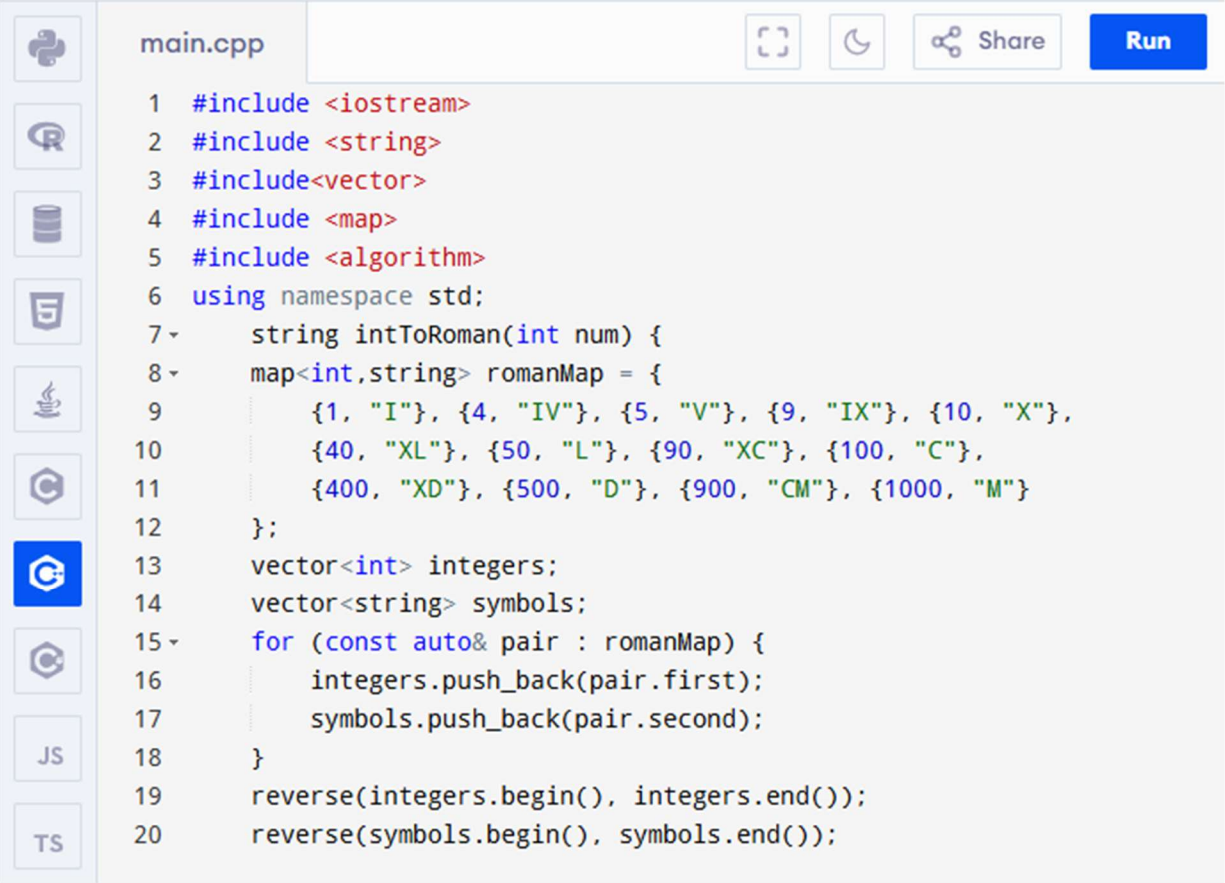
6. Write a program to convert a given year to its Roman equivalent i.e.,

| Decimal | Roman | Decimal | Roman |
|---|---|---|---|
| 1 | i | 100 | c |
| 5 | v | 500 | d |
| 10 | x | 1000 | m |
| 50 | l | | |

**e.g. 1988 = mdcccclxxxviii**

**1525 = mdxxv**

## Solution:




```cpp
#include <iostream>
#include <string>
#include<vector>
#include <map>
#include <algorithm>
using namespace std;
    string intToRoman(int num) {
    map<int,string> romanMap = {
        {1, "I"}, {4, "IV"}, {5, "V"}, {9, "IX"}, {10, "X"},
        {40, "XL"}, {50, "L"}, {90, "XC"}, {100, "C"},
        {400, "XD"}, {500, "D"}, {900, "CM"}, {1000, "M"}
    };
    vector<int> integers;
    vector<string> symbols;
    for (const auto& pair : romanMap) {
        integers.push_back(pair.first);
        symbols.push_back(pair.second);
    }
    reverse(integers.begin(), integers.end());
    reverse(symbols.begin(), symbols.end());
```



I hear and I forget. I see and I remember. I do and I understand.

Confucius

main.cpp

Share Run

```cpp

    string romanNum = "";
    for (size_t i = 0; i < integers.size(); ++i) {
        while (num >= integers[i]) {
            romanNum += symbols[i];
            num -= integers[i];
        }
    }
    return romanNum;
}
int main() {
    int year;
    cout << "Enter a year to convert to Roman numerals: ";
    cin >> year;
if (year >= 1 && year <= 3999) {
    string romanEquivalent = intToRoman(year);
    cout << "The Roman equivalent of " << year << " is: " <<
        romanEquivalent << endl;
    } else {
    cout << "Please enter a year between 1 and 3999." << endl;
     }
     return 0;
}
```

**Result:**

Output

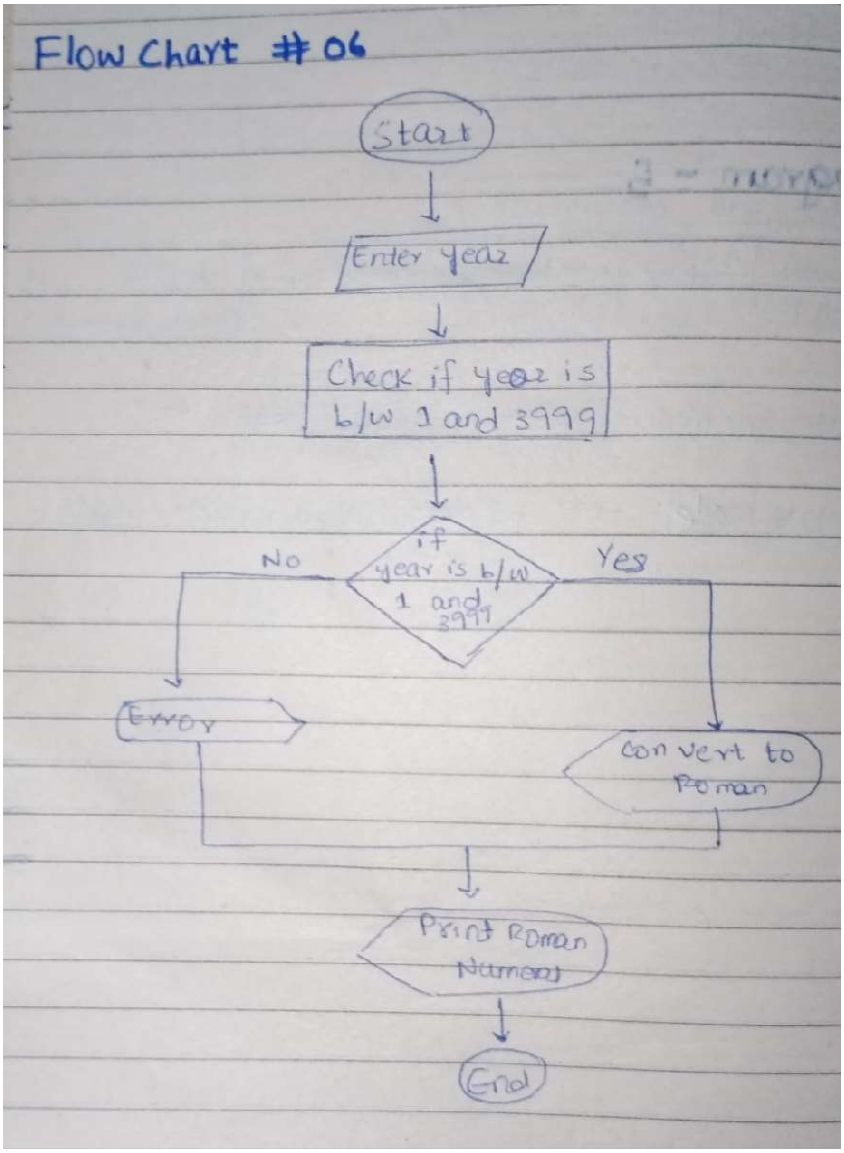
```
Enter a year to convert to Roman numerals: 2006
The Roman equivalent of 2006 is: MMVI


=== Code Execution Successful ===
```



I hear and I forget. I see and I remember. I do and I understand.

Confucius

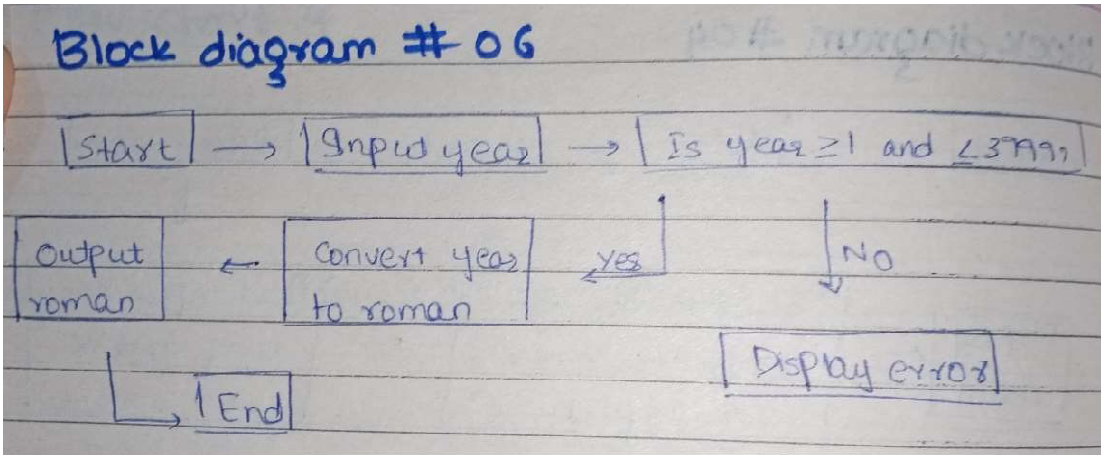
**Flowchart:**




 I hear and I forget. I see and I remember. I do and I understand.

Confucius

**Block diagram:**




 I hear and I forget. I see and I remember. I do and I understand.

Confucius