Samir Khadka (19701ks)
CS360:Individual Assignment: Assignment#1

//Question 1

```
#include <iostream>
#include <cmath>
#include <string>
#include <sstream>

class Complex {
private:
    double real;
    double imaginary;

public:
      Complex() : real(0), imaginary(0) {}

      Complex(double real, double imaginary) : real(real), imaginary(imaginary) {}

     Complex(const std::string& complexStr) {
       std::stringstream ss(complexStr);
       char delim;
       ss >> real >> delim >> imaginary;
    }

      double getReal() const {
       return real;
    }

     double getImaginary() const {
       return imaginary;
    }

     double magnitude() const {
       return sqrt(real * real + imaginary * imaginary);
    }


    double angle() const {
       return atan2(imaginary, real);
    }

      Complex conjugate() const {
```

```
        return Complex(real, -imaginary);
    }

    Complex operator+(const Complex& other) const {
        return Complex(real + other.real, imaginary + other.imaginary);
    }

    Complex operator-(const Complex& other) const {
        return Complex(real - other.real, imaginary - other.imaginary);
    }

    Complex operator*(const Complex& other) const {
        double newReal = real * other.real - imaginary * other.imaginary;
        double newImaginary = real * other.imaginary + imaginary * other.real;
        return Complex(newReal, newImaginary);
    }

    Complex operator/(const Complex& other) const {
        double divisor = other.real * other.real + other.imaginary * other.imaginary;
        double newReal = (real * other.real + imaginary * other.imaginary) / divisor;
        double newImaginary = (imaginary * other.real - real * other.imaginary) / divisor;
        return Complex(newReal, newImaginary);
    }

    void print() const {
        std::cout << "(" << real << ", " << imaginary << ")\n";
    }
};

int main() {
    Complex a(3, 4);
    Complex b("5, 6");

    Complex sum = a + b;
    Complex difference = a - b;
    Complex product = a * b;
    Complex quotient = a / b;

    std::cout << "Sum: ";
    sum.print();
    std::cout << "Difference: ";
    difference.print();
    std::cout << "Product: ";
    product.print();
```

```
    std::cout << "Quotient: ";
    quotient.print();

    return 0;
}

//Question 2

#include <vector>
#include <string>
#include <sstream>
#include <iostream>

class Matrix {
private:
    std::vector<std::vector<int>> data;
    bool isNaM;

public:

    Matrix(const std::string& input) {
        std::stringstream ss(input);
        int num;
        std::vector<int> row;

        while (ss >> num) {
            row.push_back(num);
            if (ss.peek() == ',') ss.ignore();
            if (ss.peek() == ';') {
                data.push_back(row);
                row.clear();
                ss.ignore();
            }
        }
        data.push_back(row);


        int cols = data[0].size();
        isNaM = false;
        for (const auto& r : data) {
            if (r.size() != cols) {
                isNaM = true;
                break;
            }
```

```
        }
    }


    ~Matrix() {

        data.clear();
    }


    bool IsNaM() const { return isNaM; }


    int& operator()(int i, int j) { return data[i][j]; }
    const int& operator()(int i, int j) const { return data[i][j]; }


};

int main() {

    Matrix m("1,2,3;4,5,6;7,8,9");


    if (m.IsNaM()) {
        std::cout << "The matrix is not valid.\n";
    } else {
        std::cout << "The matrix is valid.\n";
    }


    std::cout << "The element at (0, 0) is " << m(0, 0) << "\n";

    return 0;
}
```