Ronish Shrestha
19707
CS360_Assignment4

Q1)

```cpp
#include <iostream>
#include <iomanip> // Include <iomanip> for setw
#include <stdexcept>

class DoubleSubscriptedArray {
    friend std::ostream &operator<<(std::ostream &, const DoubleSubscriptedArray &);
    friend std::istream &operator>>(std::istream &, DoubleSubscriptedArray &);
public:
    explicit DoubleSubscriptedArray(int = 3, int = 5); // default constructor
    DoubleSubscriptedArray(const DoubleSubscriptedArray &); // copy constructor
    ~DoubleSubscriptedArray(); // destructor
    DoubleSubscriptedArray &operator=(const DoubleSubscriptedArray &); // assignment operator
    bool operator==(const DoubleSubscriptedArray &) const; // equality operator
    bool operator!=(const DoubleSubscriptedArray &right) const {
        return !(*this == right);
    }
    // function operator() overloads
    int &operator()(int, int); // return int& for lvalue
    int operator()(int, int) const; // return int for rvalue
private:
```

Output:
Zero-indexing used for both rows and columns

```
Enter values for the chessBoard (3x5):
1
2
3
4
5
6
7
8
9
3
4
2
1
5
6

Output chessBoard (3x5):
    1    2    3    4    5
    6    7    8    9    3
    4    2    1    5    6

Accessing element at row 1 and column 3: 9
```

Q2)

```cpp
#include <iostream>
#include <vector>
#include <algorithm>

class Term {
public:
    Term(double coef = 0.0, int exp = 0) : coefficient(coef), exponent(exp) {}

    double getCoefficient() const { return coefficient; }
    int getExponent() const { return exponent; }

    void setCoefficient(double coef) { coefficient = coef; }
    void setExponent(int exp) { exponent = exp; }

    bool operator==(const Term& other) const {
        return exponent == other.exponent;
    }

private:
    double coefficient;
    int exponent;
};

class Polynomial {
public:
    // Constructors
```

Output:

```
Polynomial 1: 2x^4 - 3x^1 + 6
Polynomial 2: 5x^2 + 2
Sum: 2x^4 + 5x^2 - 3x^1 + 8
Difference: 2x^4 - 5x^2 - 3x^1 + 4
Product: 10x^6 + 4x^4 - 15x^3 + 30x^2 - 6x^1
+ 12
```