



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ «Информатика и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Лабораторная работа № 7
по курсу «Языки и методы программирования»
«Разработка простейшего класса на C++»

Студент группы ИУ9-22Б Тараканов В. Д.

Преподаватель Посевин Д. П.

Москва 2024

1 Задание

Целочисленная матрица размером $m \times n$ с операциями: 1. получение количества строк; 2. получение количества столбцов; 3. получение ссылки на указанный элемент; 4. перестановка двух строк; 5. перестановка двух столбцов.

2 Результаты

Листинг 1 — Объявление класса IntegerMatrix

```
1
2  #ifndef DECLARATION_H
3  #define DECLARATION_H
4
5  class IntegerMatrix {
6      private:
7          int **matrix;
8          int rows;
9          int cols;
10
11     public:
12         IntegerMatrix(int m, int n);
13         IntegerMatrix(IntegerMatrix &other);
14
15         virtual ~IntegerMatrix();
16         IntegerMatrix& operator=(IntegerMatrix &other);
17
18         int getRows() const;
19         int getCols() const;
20         int& getElement(int row, int col);
21
22         void swapRows(int row1, int row2);
23         void swapCols(int col1, int col2);
24     };
25
26 #endif
27
```

Листинг 2 — Реализация методов класса Polynomial

```
1
2     #include "declaration.h"
3
4     IntegerMatrix::IntegerMatrix(int m, int n) : rows(m), cols(n) {
5         matrix = new int*[rows];
6         for (int i = 0; i < rows; ++i) {
7             matrix[i] = new int[cols];
8             for (int j = 0; j < cols; ++j) {
9                 matrix[i][j] = 0;
10            }
11        }
12    }
13
14    IntegerMatrix::IntegerMatrix(IntegerMatrix &other) : rows(other.
rows), cols(other.cols) {
15        matrix = new int*[rows];
16        for (int i = 0; i < rows; ++i) {
17            matrix[i] = new int[cols];
18            for (int j = 0; j < cols; ++j) {
19                matrix[i][j] = other.matrix[i][j];
20            }
21        }
22    }
23
24
25    IntegerMatrix::~~IntegerMatrix() {
26        for (int i = 0; i < rows; ++i) {
27            delete [] matrix[i];
28        }
29        delete [] matrix;
30    }
31
32
33
```

Листинг 3 — Реализация методов класса Polynomial(продолжение)

```
1
2     IntegerMatrix& IntegerMatrix::operator=(IntegerMatrix &other) {
3         if (this != &other) {
4
5             for (int i = 0; i < rows; ++i) {
6                 delete[] matrix[i];
7             }
8             delete[] matrix;
9
10            rows = other.rows;
11            cols = other.cols;
12            matrix = new int*[rows];
13            for (int i = 0; i < rows; ++i) {
14                matrix[i] = new int[cols];
15                for (int j = 0; j < cols; ++j) {
16                    matrix[i][j] = other.matrix[i][j];
17                }
18            }
19        }
20        return *this;
21    }
22
23    int IntegerMatrix::getRows() const {
24        return rows;
25    }
26
27    int IntegerMatrix::getCols() const {
28        return cols;
29    }
30
31    int& IntegerMatrix::getElement(int row, int col) {
32        return matrix[row][col];
33    }
34
35    void IntegerMatrix::swapRows(int row1, int row2) {
36        if (row1 >= 0 && row1 < rows && row2 >= 0 && row2 < rows) {
37            int *temp = matrix[row1];
38            matrix[row1] = matrix[row2];
39            matrix[row2] = temp;
40        }
41    }
42
43    void IntegerMatrix::swapCols(int col1, int col2) {
44        if (col1 >= 0 && col1 < cols && col2 >= 0 && col2 < cols) {
45            for (int i = 0; i < rows; ++i) {
46                int temp = matrix[i][col1];
47                matrix[i][col1] = matrix[i][col2];
48                matrix[i][col2] = temp;
49            }
50        }
51    }
52
53
```

Листинг 4 — Функция main для проверки работы класса IntegerMatrix

```
1      #include <iostream>
2      #include "implementation.cpp"
3
4      using namespace std;
5
6      void printMatrix(IntegerMatrix &matrix) {
7          for (int i = 0; i < matrix.getRows(); ++i) {
8              for (int j = 0; j < matrix.getCols(); ++j) {
9                  cout << matrix.getElement(i, j) << " ";
10             }
11             cout << endl;
12         }
13     }
14
15     int main() {
16         IntegerMatrix mat(3, 3);
17
18         int value = 1;
19         for (int i = 0; i < mat.getRows(); ++i) {
20             for (int j = 0; j < mat.getCols(); ++j) {
21                 mat.getElement(i, j) = value++;
22             }
23         }
24
25         cout << "Original matrix:" << endl;
26         printMatrix(mat);
27
28         IntegerMatrix copiedMat = mat;
29         cout << "\nCopied matrix:" << endl;
30         printMatrix(copiedMat);
31
32         IntegerMatrix assignedMat(2, 2);
33         assignedMat = mat;
34         cout << "\nAssigned matrix:" << endl;
35         printMatrix(assignedMat);
36         cout << "\nSwaped cols 1 and 2"<<endl;
37         mat.swapCols(1,2);
38         printMatrix(mat);
39         return 0;
40     }
41
```

Результат запуска представлен на рисунках 1.

```
Original matrix:
1 2 3
4 5 6
7 8 9

Copied matrix:
1 2 3
4 5 6
7 8 9

Assigned matrix:
1 2 3
4 5 6
7 8 9

Swaped cols 1 and 2
1 3 2
4 6 5
7 9 8
```

Рис. 1 — Результат