



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 № 10
по курсу «Языки и методы программирования»
«Реализация итераторов на языке C++»

Студент группы ИУ9-22Б Тараканов В. Д.

Преподаватель Посевин Д. П.

Москва 2024

1 Задание

Матрица размера $m \times n$ с константным однонаправленным итератором по всем возможным подматрицам (подматрица получается из матрицы удалением произвольных строк и/или столбцов).

2 Результаты

Листинг 1 — Вспомогательный класс Matrix

```
1  #ifndef MATRIX_H
2  #define MATRIX_H
3
4  #include <vector>
5  #include "ConstSubmatrixIterator.h"
6
7  class Matrix {
8  public:
9      Matrix(size_t rows, size_t cols);
10
11      size_t rowCount() const;
12      size_t colCount() const;
13
14      const int& operator()(size_t row, size_t col) const;
15
16      ConstSubmatrixIterator begin() const;
17      ConstSubmatrixIterator end() const;
18
19  private:
20      std::vector<std::vector<int>>> data;
21  };
22
23 #endif // MATRIX_H
24
```

Листинг 2 — Реализация методов класса Matrix

```
1      #include "Matrix.h"
2
3      Matrix::Matrix(size_t rows, size_t cols) : data(rows, std::vector<
int>(cols)) {}
4
5      size_t Matrix::rowCount() const {
6          return data.size();
7      }
8
9      size_t Matrix::colCount() const {
10         return data[0].size();
11     }
12
13     const int& Matrix::operator()(size_t row, size_t col) const {
14         return data[row][col];
15     }
16
17     ConstSubmatrixIterator Matrix::begin() const {
18         return ConstSubmatrixIterator(*this);
19     }
20
21     ConstSubmatrixIterator Matrix::end() const {
22         return ConstSubmatrixIterator(*this, true);
23     }
24
25
```

Листинг 3 — Класс ConstSubmatrixIterator

```
1  #ifndef CONSTSUBMATRIXITERATOR_H
2  #define CONSTSUBMATRIXITERATOR_H
3
4  #include "Matrix.h"
5
6  class ConstSubmatrixIterator {
7  public:
8      using container_type = std::vector<std::vector<int>>>;
9
10     ConstSubmatrixIterator(const Matrix& mat, bool end = false);
11
12     container_type operator*() const;
13
14     ConstSubmatrixIterator& operator++();
15     bool operator!=(const ConstSubmatrixIterator& other) const;
16
17     private:
18     const Matrix& matrix;
19     size_t rowIndex;
20     size_t colIndex;
21     bool done;
22
23     void advance();
24 };
25
26 #endif // CONSTSUBMATRIXITERATOR_H
27
28
```

Листинг 4 — Реализация методов класса ConstSubmatrixIterator

```
1      #include "ConstSubmatrixIterator.h"
2
3      ConstSubmatrixIterator::ConstSubmatrixIterator(const std::vector<
std::vector<int>>& mat, bool end)
4      : matrix(mat), rowIndex(0), colIndex(0), done(end) {
5          if (matrix.rowCount() == 0 || matrix.colCount() == 0) {
6              done = true;
7          }
8      }
9
10     ConstSubmatrixIterator::container_type ConstSubmatrixIterator::
operator*() const {
11         container_type submatrix;
12         for (size_t i = 0; i < matrix.rowCount(); ++i) {
13             if (i != rowIndex) {
14                 std::vector<int> row;
15                 for (size_t j = 0; j < matrix.colCount(); ++j) {
16                     if (j != colIndex) {
17                         row.push_back(matrix(i, j));
18                     }
19                 }
20                 submatrix.push_back(row);
21             }
22         }
23         return submatrix;
24     }
25
26     ConstSubmatrixIterator& ConstSubmatrixIterator::operator++() {
27         advance();
28         return *this;
29     }
30
31     bool ConstSubmatrixIterator::operator!=(const
ConstSubmatrixIterator& other) const {
32         return done != other.done;
33     }
34
35     void ConstSubmatrixIterator::advance() {
36         if (++colIndex >= matrix.colCount()) {
37             colIndex = 0;
38             if (++rowIndex >= matrix.rowCount()) {
39                 done = true;
40             }
41         }
42     }
43
44
```

Листинг 5 — Функция main, проверяющая работу класса ConstSubmatrixIterator

```
1  #include <iostream>
2  #include "Vector.h"
3  #include "VectorSequence.h"
4
5  int main() {
6      VectorSequence sequence;
7      sequence.addVector(Vector({1.0, 1.0, 1.0}));
8      sequence.addVector(Vector({1.0, 1.0, 1.0}));
9      sequence.addVector(Vector({7.0, 8.0, 9.0}));
10
11     for (auto it = sequence.begin(); it != sequence.end(); ++it) {
12         std::cout << *it << " ";
13     }
14
15
16     return 0;
17 }
18
19
```

Результат запуска представлен на рисунке 1.

```
1  ----
2
3  ----
4
5  ----
6
7  ----
8
9  ----
10
11 1 2
12 4 5
13  ----
14 2 3
15 5 6
16  ----
17 4 5
18 7 8
19  ----
20 5 6
21 8 9
22  ----
23 1 2 3
24 4 5 6
25 7 8 9
26  ----
```

Рис. 1 — Результат