



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 № 8**  
**по курсу «Языки и методы программирования»**  
**«Разработка шаблона класса»**

Студент группы ИУ9-22Б Федуков А. А.

Преподаватель Посевин Д. П.

8 мая 2024 г.

# Цель работы

Целью данной работы является изучение шаблонов классов языка C++

## Задание

Согласно выбранному из таблиц 1–16 описанию требуется составить шаблон класса, разместив его в отдельном заголовочном файле. Проверку работоспособности класса требуется организовать в функции `main`, размещённой в файле «`main.cpp`».

## Задание 1

`PascalArray<int L,int R,class T>` – массив с элементами типа `T`, индексируемые от `L` до `R`, с перегруженной операцией индексации и операцией конкатенации двух массивов, которая допустима в случае, если правая граница первого массива на единицу меньше левой границы второго массива. Массив, в котором  $R = L - 1$ , считается пустым и не обладает операцией индексации.

## Реализация

Я описал класс в файле заголовков [PascalArray.h](#)

Сгенерировал его экземпляры и проверил работоспособность необходимых функций уже в [main.cpp](#)

## Код

Листинг 1: Файл `PascalArray.h`

```
1 #include <stdexcept>
2 #include <iostream>
3 #define DEBUGINFO 0
4
5 template <int L, int R, class T>
6 class PascalArray
7 {
```

```

8 public :
9     T *data;
10
11     PascalArray()
12     {
13         DEBUGINFO ? std::cout << "make " << L << " " << R << std::endl :
            std::cout;
14         data = new T[R - L + 1];
15     }
16
17     PascalArray(PascalArray<L, R, T> &obj)
18     {
19         DEBUGINFO ? std::cout << "COPY " << L << " " << R << std::endl :
            std::cout;
20         T *l = new T[R - L + 1];
21         for (size_t i = 0; i < R - L + 1; i++)
22         {
23             l[i] = obj.data[i];
24         }
25         this->data = l;
26     }
27     ~PascalArray()
28     {
29         DEBUGINFO ? std::cout << "delete " << L << " " << R << std::endl
            : std::cout;
30         delete [] data;
31     }
32     T &operator [] (unsigned i) const
33     {
34         DEBUGINFO ? std::cout << "index " << i << " of " << L << " " <<
            R << std::endl : std::cout;
35
36         if (i > R || i < L || R == L - 1)
37         {
38             throw std::out_of_range("Index error of PascalArray!");
39         }
40         else
41         {
42             return data[i - L];
43         }
44     }
45     template <int L2, int R2, class T2>
46     PascalArray<L, R2, T> operator+(const PascalArray<L2, R2, T2> &l2)
47     {
48         if (R == L2 - 1)
49         {

```

```

50         PascalArray<L, R2, T> l;
51
52         for (size_t i = 0; i < R - L + 1; i++)
53         {
54             l.data[i] = data[i];
55         }
56         for (size_t i = 0; i < R2 - L2 + 1; i++)
57         {
58             l.data[R - L + 1 + i] = l2.data[i];
59         }
60
61         // return PascalArray<L, R2, T>(l);
62         return l;
63     }
64     else
65     {
66         throw std::out_of_range("Bad R1 and L2 for operation +");
67     }
68 }
69 };

```

## Листинг 2: Файл main.cpp

```

1 #include "PascalArray.h"
2 #include <iostream>
3
4 int main()
5 {
6     PascalArray<10, 20, int> l1;
7     l1[10] = 1;
8     l1[11] = 2;
9     PascalArray<21, 22, int> l2;
10    l2[21] = 3;
11    std::cout << "l1[10]: " << l1[10] << std::endl;
12    std::cout << "l2[21]: " << l2[21] << std::endl;
13    PascalArray<10, 22, int> l3 = l1 + l2;
14    std::cout << "l3[21]: " << l3[21] << std::endl;
15    std::cout << "(l1+l2)[21]: " << (l1+l2)[21] << std::endl;
16    return 0;
17 }

```

## Вывод программы

Программа создала экземпляры класса и протестировала все заявленные операции

Листинг 3: Вывод программы

```
1 11 [10]: 1
2 12 [21]: 3
3 13 [21]: 3
4 (11+12) [21]: 3
```

## Вывод

Выполняя эту лабораторную работу, я научился использовать шаблоны в C++. Было очень непонятно, но я справился!