

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ  
УНИВЕРСИТЕТ им. В. Г. ШУХОВА» (БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения вычислительной техники и  
автоматизированных систем

## **Лабораторная работа №8**

по дисциплине: «ООП»

Выполнил: ст. группы ПВ-211

Чувилко Илья Романович

Проверил:

Буханов Дмитрий Геннадьевич

Харитонов Сергей Дмитриевич

Белгород 2023 г.

**Тема:** Создание шаблонов классов в C++.

**Вариант:** 23

**Цель работы:** Получение теоретических знаний о шаблонах классов в C++.  
Получение практических навыков по созданию классов-шаблонов C++.

**Задание:**

1. Изучить теоретические сведения о шаблонах классов в C++.
2. Разработать программу в соответствии с заданным вариантом задания
3. Оформить отчет.

**Выполнение работы:**

**Задание 1:** Реализовать шаблон класса в соответствии с указанным вариантом.  
Предусмотреть необходимые методы для работы со структурой данных, указанной в варианте. Предусмотреть исключительные ситуации, которые могут возникнуть в процессе работы.

- Стек

```
#include <iostream>
#include <cassert>
#include "string"

using namespace std;

const int sizeA = 100;

template<typename T>
struct Stack {
    T data[sizeA];
    unsigned ptr;

    Stack() {
        ptr = 0;
    }

    [[nodiscard]] bool EmptyStack() const {
        return ptr == 0;
    }

    void Push(T x) {
        assert(ptr <= sizeA);
        data[ptr++] = x;
    }

    void Pop() {
        assert(!EmptyStack());
        ptr--;
    }

    T Get() {
        return data[ptr - 1];
    }
};
```

**Задание 2:** Дан файл содержащий текст, который представляет собой различные предложения. Считать предложения из текста в разработанную структуру данных таким образом, чтобы первыми словами были существительные, далее глаголы, затем все остальное. Для получения информации к какой части речи принадлежит слово использовать справочники, хранящиеся в других файлах(использовать те же разработанные шаблоны).

```
void shredder(string &a, string *shred, int &counter) {
    string tmp;
    for (auto i: a) {
        if ((i == ' ' || i == ',') && tmp.size() > 0) {
            shred[counter++] = tmp;
            tmp = "";
        } else
            tmp.push_back(i);
    }
    if (tmp.size() > 0)
        shred[counter++] = tmp;
}

void finder(Stack<string> &vvv, string *shred, int &counter, string *dict) {
    for (int i = 0; i < counter; i++)
        for (int j = 0; j < dict->size(); j++)
            if (dict[j] == shred[i]) {
                vvv.Push(shred[i]);
                shred[i] = "";
            }
}

void sort(string &a, Stack<string> &vvv) {
    string state[] = {"Panama"};
    string noun[] = {"plan", "man", "canal"};
    int counter = 0;
    string shred[100];
    shredder(a, shred, counter);
    finder(vvv, shred, counter, state);
    finder(vvv, shred, counter, noun);
    for (int i = 0; i < counter; i++)
        if (!shred[i].empty()) {
            vvv.Push(shred[i]);
            shred[i] = "";
        }
}

int main() {
    string a = "a man, a plan a canal Panama";
    Stack<string> vvv;
    sort(a, vvv);
    while (!vvv.EmptyStack()) {
        cout << vvv.Get() << ' ';
        vvv.Pop();
    }
    return 0;
}
```

**Вывод:** в ходе лабораторной работы мы получили теоретические знания о шаблонах классов в C++, а также практические навыки по созданию классов-шаблонов C++