ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»

(СПбГУТ)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Факультет Информационных технологий и программной инженерии

Кафедра Систем обработки данных

**Дисциплина «Технологии программирования»**

**ОТЧЕТ**

**за практическое занятие №7**

**Тема: Реализация шаблона класса**

Выполнил

Студент 2 курса, гр. ИСТ-422

\_\_\_\_\_\_\_\_\_\_\_\_\_ Ващук А.А.

Принял

Доцент кафедры БИС

\_\_\_\_\_\_\_\_\_\_\_\_\_ Медведев В.А.

Санкт-Петербург

2025

**Тема и цель практического занятия:**

Получение практических навыков разработки и использования шаблонов классов для хранения и обработки данных.

**Задание на практическое занятие:**

*Выберите один из вариантов задания.*

**Вариант А.**

**Задание 1.** В соответствии с *индивидуальным заданием* (см. приложение) создать класс для предметной области.

**Задание 2.** Спроектировать шаблон класса с одним обобщенным типом.

**Задание 3.** Разработать реализацию шаблона класса для каждого из двух вычисляемых показателей.

**Текст программы:**

#include <iostream>

#include <vector>

#include <stdexcept>

#include <string>

#include "../include/DoxTemplate.h"

#include "../include/Dox.h"

int main() {

    // Создание процессора для документов

    DoxxTemplate<Doxx> processor;

    // Добавление документов

    processor.addDocument(Doxx("Contract", "20220101"));

    processor.addDocument(Doxx("Invoice", "20230101"));

    processor.addDocument(Doxx("Report", "20210115"));

    processor.addDocument(Doxx("Invoice", "20230520"));

    processor.addDocument(Doxx("Contract", "20221231"));

    // Показатель 1: Самый ранний документ

    try {

        Doxx earliest = processor.getEarliestDocument();

        std::cout << "Earliest document: Type = " << earliest.getType()

                  << ", Date = " << earliest.getDate() << std::endl;

    } catch (const std::exception& e) {

        std::cerr << "Error: " << e.what() << std::endl;

    }

    // Показатель 2: Количество документов типа "Invoice"

    int invoiceCount = processor.countByType("Invoice");

    std::cout << "Number of Invoices: " << invoiceCount << std::endl;

    return 0;

}

#pragma once

#include <string>

class Doxx {

private:

    std::string type;

    std::string date; // Дата в формате "ГГГГММДД"

public:

    Doxx(const std::string& t, const std::string& d) : type(t), date(d) {}

    std::string getType() const { return type; }

    std::string getDate() const { return date; }

};

#pragma once

#include <vector>

#include <stdexcept>

#include <string>

template <class T>

class DoxxTemplate {

private:

    std::vector<T> docs; // Используем обобщенный тип T

public:

    // Добавление документа

    void addDocument(const T& doc) {

        docs.push\_back(doc);

    }

    // Поиск документа с самой ранней датой

    T getEarliestDocument() const {

        if (docs.empty()) {

            throw std::logic\_error("No documents to process.");

        }

        T earliest = docs[0];

        for (const auto& doc : docs) {

            if (doc.getDate() < earliest.getDate()) {

                earliest = doc;

            }

        }

        return earliest;

    }

    // Подсчёт документов определённого типа

    int countByType(const std::string& targetType) const {

        int count = 0;

        for (const auto& doc : docs) {

            if (doc.getType() == targetType) {

                count++;

            }

        }

        return count;

    }

};

**Выводы по созданному проекту и использованию средств языка программирования:**

В программе были задействованы шаблоны функций (template <class T>), а это значит, что класс может работать с любым типом данных.

Был создан класс для документа (Doxx), и шаблон класса (DoxxTemplate).

Реализован поиск и подсчет доков определенного типа.