**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное ГОСУДАРСТВЕННОЕ бюджетное ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информатики

и вычислительной техники

Кафедра информационной

безопасности

**ОТЧЕТ**

**По Лабораторной работе №2**

Выполнил:

студент гр. БИс-32

Фомин Е.В.

Проверил:

Ассистент кафедры ИБ

Ситников И.В.

Йошкар-Ола

2017

**Ход работы**

**Цель работы:** Реализовать класс MyString, который будет являться аналогом класса String языка C#.

**Требования к программе:**

1. В качестве поля класс должен содержать не переменную string, а массив символов (char []).
2. Класс должен реализовывать перегрузку операторов (например, +, \*, >, <и т.д.). Подумайте, какие операторы действительно потребуются вашему классу.
3. Класс должен содержать несколько конструкторов с различными принимаемыми параметрами.
4. Класс должен содержать несколько методов для работы с массивом символов, так, если бы он представлял собой строку (например, взятие подстроки, замена символов в строке и т.д.). Реализуйте 4-5 методов на ваш выбор. Также, предусмотрите по 1-2 перегрузки каждого метода.
5. Класс должен содержать перегрузки операций преобразования типов. Подумайте, какие перегрузки должны быть неявными, а какие должны требовать явного указания типа.
6. **Класс НЕ должен использовать методы, уже реализованные в классе String.**

**Теоретические сведения:**

**Перегрузка методов**– это объявление в классе методов с одинаковыми именами при этом с различными параметрами.

**Применение**

Наследование является механизмом повторного использования кода и способствует независимому расширению программного обеспечения через открытые классы и интерфейсы Установка отношения наследования между классами порождает иерархию классов

**Наследование классов** — очень мощная возможность в ООП. Оно позволяет создавать производные классы взяв за основу все методы и элементы базового класса Таким образом экономится масса времени на написание и отладку кода новой программы. Объекты производного класса свободно могут использовать всё, что создано и отлажено в базовом классе. При этом, мы можем в производный класс, дописать необходимый код для усовершенствования программы: добавить новые элементы, методы и т.д.. Базовый класс останется нетронутым. Эту тему вполне возможно освоить новичкам. Необходимо только познакомиться с синтаксисом и некоторыми особенностями. Ниже приведен простой код программы, который мы детально разберем под листингом. В этой программе созданы два класса: базовый — FirstClass и производный от него SecondClass.

Ссылка на GitHub: <https://github.com/KPSS322/Pna9vy2>

# Вывод

Входе выполнения работы, был разработан класс MyString, который является аналогом класса String в C#. Реализованы методы поиска подстроки в строке с перегрузкой; метод поиска символа в строке с перегрузкой; метод вычисления размера; метод копирования из одной строки и вставка её на место другой строки, с перегрузкой; логический метод сравнения идентичности строк; и метод объединения строк с перегрузкой.