

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное
учреждение высшего образования

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Инженерная школа информационных технологий и робототехники
Отделение информационных технологий
Направление информатика и вычислительная техника

Отчет
по лабораторной работе № 3

по дисциплине
«ПРОГРАММИРОВАНИЕ»

Вариант № 26

Работа с базовым контейнером string на языке программирования C++.

Выполнил:

Студент группы 8В32

Д.О.Карташов

Проверил:

Ассистент ОИТ ИШИТР

А.Ю.Малкин

Томск 2024

Цель работы

Получить навыки работы с базовым контейнером `string` на языке программирования C++.

Задание

Выполнить 3 задания, которые представлены ниже. Проект для выполнения заданий собрать и отладить на языке программирования C++ с применением CMake в VS Code.

1. Даны `string s` и `vector<string> words`, где каждый элемент вектора – строка, состоящая из строчных латинских букв. Все строки – только из строчных латинских букв. Написать функцию, которая возвращает `true`, если `s` является акронимом для `words`. Это происходит в том случае, если мы соединяем первый символ каждого элемента вектора `words` по порядку и получаем строку `s`. Например, строка "тпу" может быть сформирована из вектора ["томский", "политехнический", "университет"], но не из вектора ["томский", "университет", "политехнический"].

2. Написать функцию, проверяющую, что вводимая пользователем строка, которая состоит из строчных латинских букв и пробелов, является палиндромом (читается вперёд и назад одинаково). Строка может содержать произвольное количество пробелов подряд; таким образом, для "never odd or even" программа должна вернуть `true`.

3. Даны две строки: `string1` и `string2`, состоящие из строчных латинских букв. За одно действие можно добавить один символ или к `string1`, или к `string2`. Написать функцию, которая возвращает минимальное необходимое количество действий, чтобы `string1` и `string2` стали анаграммами по отношению друг к другу. Две строки будут анаграммами друг для друга, если они состоят из одинакового набора букв, но порядок может быть различным.

Ход работы

На рисунке 1 демонстрируется результат выполнения функции `isAcronym()` с аргументами "tpu" и ["tomsk", "polytechnic", "university"]. Получено значение `true`.

```
100 | std::cout << isAcronym("tpu", {"tomsk", "polytechnic", "university"}) << std::endl;
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
true
```

Рисунок 1 – "tpu" является акронимом ["tomsk", "polytechnic", "university"]

На рисунке 2 демонстрируется результат выполнения функции isAcronym() с аргументами "tpu" и ["tomsk", "state", "university"]. Получено значение false.

```
101 | std::cout << isAcronym("tpu", {"tomsk", "state", "university"}) << std::endl;
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
false
```

Рисунок 2 – "tpu" не является акронимом ["tomsk", "state", "university"]

На рисунке 3 демонстрируется результат выполнения функции isPalindrom() с аргументом "never odd or even". Получено значение true.

```
103 | std::cout << isPalindrom("never odd or even") << std::endl;
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
true
```

Рисунок 3 – "never odd or even" является палиндромом

На рисунке 4 демонстрируется результат выполнения функции isPalindrom() с аргументом "never even or odd". Получено значение false.

```
104 | std::cout << isPalindrom("never even or odd") << std::endl;
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
false
```

Рисунок 4 – "never even or odd" не является палиндромом

На рисунке 5 демонстрируется результат выполнения функции maxScore() с аргументами ["ab", "ac", "bc"], ['a', 'b', 'c'], [1, 3, 2]. Получено значение 5.

```
106 | std::cout << maxScore({"ab", "ac", "bc"}, {'a', 'b', 'c'}, {1, 3, 2}) << std::endl;  
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
5
```

Рисунок 5 – Слово "bc" даёт наибольший счёт – 5

На рисунке 6 демонстрируется результат выполнения функции `maxScore()` с аргументами `["ab", "ac", "bc"]`, `['x', 'y', 'z']`, `[2, 3, 1]`. Получено значение -1.

```
107 | std::cout << maxScore({"ab", "ac", "bc"}, {'x', 'y', 'z'}, {2, 3, 1}) << std::endl;  
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS  
-1
```

Рисунок 6 – Из данного набора букв нельзя составить данные слова

Вывод

Получены навыки работы с базовым контейнером `string` на языке программирования C++.

Приложение 1 – Код программы main.cpp

Ссылка на исполняемый код: *<https://github.com/The-Aozzi/Lab3>*