

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
Національний технічний університет України
Факультет інформатики та обчислювальної техніки
«Київський політехнічний інститут імені Ігоря Сікорського»
Кафедра інформаційних систем та технологій

Звіт
з лабораторної роботи № 8
«Робота з рядками мовою C++»
з дисципліни
«Програмування – 1. Основи програмування»

Варіант № 25

Перевірів:

доц. Корнага Ярослав Ігорович

Виконала: Павлова Софія

Студентка гр. ІС-12 , ФІОТ

1 курс,

залікова книжка № ІС-1224

Лабораторна робота № 8

Тема: Робота з рядками мовою C++.

Мета: Ознайомитися з особливостями роботи з рядками, опанувати технологію застосування та обробки рядків, навчитися розробляти алгоритми та програми із застосуванням рядків.

Обладнання: Персональні комп'ютери.

ЗАВДАННЯ

Завдання №1

1. У рядку символів визначити кількість слів. Вилучити з рядка однолітерні слова та зайві пропуски, коми, крапки. Слова відділяються один від одного довільною кількістю пропусків, ком і крапок.

Алгоритм:

1. Отримати значення рядка з зайвими пробілами, комами та крапками. Створити символний рядок.
2. Видалити зайві пробіли, коми, крапки.
 - 2.1. Якщо рядок починається з пробілу, коми або крапки, прибрати їх.
 - 2.2. Якщо рядок закінчується пробілом або комою, прибрати їх.
 - 2.3. Якщо серед рядка йдуть 2 пробіли, 2 крапки або 2 коми підряд, залишити одну.
 - 2.4. Згенерувати відформатований рядок.
3. Видалити слова з 1 букви.
 - 3.1. Якщо наступний елемент, починаючи від 0 – не пробіл, кома або крапка, вважати, що знайдено слово. Записуємо «1» у новий символний рядок.
 - 3.2. Якщо наступний елемент – пробіл, кома або крапка, вважати, що слово завершено. Записуємо «0» у символний рядок, як у попередній дії.
 - 3.3. Якщо у новому символному рядку останні 3 цифри = «010», видаляємо два останні елементи головного рядка.
 - 3.4. У іншому разі, продовжуємо перевірку.
4. Порахувати кількість слів у головному символному рядку.
 - 4.1. Увести нову булеву змінну, що приймає «1», якщо знайдено нове слово та «0», якщо знайдено новий пробіл.
 - 4.2. Присвоїти змінній «0».
 - 4.3. Якщо наступний елемент – не пробіл, кома або крапка і змінна = «0», присвоїти змінній «1» та збільшити лічильник слів на 1.
 - 4.4. Якщо наступний елемент – пробіл, кома або крапка, присвоїти змінній «0».
5. Вивести результат.

Код:

```
#include <iostream>
#include <string>
#include <windows.h>
using namespace std;

string str;
int len; // длина строки

//===== Функция создания строки =====//
void create() {
    cout << "Введите строку: \n";
    // получаем строчку (аналог cin >>)
    getline(cin, str);
    // измеряем длину строки
    len = str.length();
}

//===== Функция удаления лишних (пробелов,?! ) =====//
void del_space() {
    int slovo; // маркер
    НАЛИЧИЯ СЛОВА

    for (int i = 0; i < str.length(); i++) {
        // если строчка начинается с (пробела,?!), убираем его и сдвигаем строчку
        на 1 СИМВОЛ ВЛЕВО
        if (str[0] == ' ' || str[0] == ',' || str[0] == '.' || str[0] == '?' || str[0] == '!') {
            // "erase()" - для удаления символов со строчки
            str.erase(0, 1);
            len = str.length();
        }
        // если 2 (пробела,?! ) подряд, убираем последний и сдвигаем строчку на 1
        СИМВОЛ ВЛЕВО
        if ((str[i] == ' ' && str[i + 1] == ' ') || (str[i] == ',' && str[i + 1] == ',') || (str[i] == '.' && str[i + 1] == '.') || (str[i] == '?' && str[i + 1] == '?') || (str[i] == '!' && str[i + 1] == '!') ||
            (str[i] == ' ' && str[i + 1] == ',') || (str[i] == ' ' && str[i + 1] == '.') || (str[i] == ' ' && str[i + 1] == '?') || (str[i] == ' ' && str[i + 1] == '!') ||
            (str[i] == ',' && str[i + 1] == ' ') || (str[i] == ',' && str[i + 1] == '.') || (str[i] == ',' && str[i + 1] == '?') || (str[i] == ',' && str[i + 1] == '!') ||
            (str[i] == '.' && str[i + 1] == ' ') || (str[i] == '.' && str[i + 1] == ',') || (str[i] == '.' && str[i + 1] == '?') || (str[i] == '.' && str[i + 1] == '!') ||
            (str[i] == '?' && str[i + 1] == ' ') || (str[i] == '?' && str[i + 1] == ',') || (str[i] == '?' && str[i + 1] == '.') || (str[i] == '?' && str[i + 1] == '!') ||
            (str[i] == '!' && str[i + 1] == ' ') || (str[i] == '!' && str[i + 1] == ',') || (str[i] == '!' && str[i + 1] == '.') || (str[i] == '!' && str[i + 1] == '?')) {
            str.erase(i, 1);
            i = (i - 1);
            len = str.length();
        }
    }
}
```

```

    }
    // если строчка заканчивается (пробелом,), убираем
    if (str[len - 1] == ' ' || str[len - 1] == ',') {
        str.erase(len - 1, 1);
        len = str.length();
    }
}
cout << "\n/===== КОНЕЧНЫЙ РЕЗУЛЬТАТ
=====^n";
cout << "УДАЛЕНЫ ЛИШНИЕ ПРОБЕЛЫ: \n" << str << "\n";
}
//===== Функция удаления слов с 1 буквы =====//
void del_w() {
    string slen; // строка и массив -
    // маркер пробелов
    string s1 = "01"; // массив для добавления "1" в slen
    string s2 = "10"; // массив для добавления "0" в slen
    int lenth; // длина строки,
    // длина массива - маркера пробелов и счётчик слов

    // пока первое слово из 1 буквы, убираем его
    while (str[1] == ' ') {
        str.erase(0, 2);
        len = str.length();
    }

    for (int i = 0; i < len; i++) {
        // если элемент - НЕ (пробел,.?! )
        if (str[i] != ' ') {
            slen.append(s1, 1);
            lenth = slen.length();
            if (i > 2) {
                if (lenth >= 3) {
                    slen.erase(0, lenth - 3);
                }
                lenth = slen.length();
            }
        }
        // если элемент - (пробел,.?! )
        if (str[i] == ' ' || str[i] == ',' || str[i] == '.' || str[i] == '?' || str[i] == '!') {
            slen.append(s2, 1);
            lenth = slen.length();
            if (i > 2) {
                slen.erase(0, lenth - 3);
                if (slen.compare("010") == 0) {
                    str.erase(i - 1, 2);
                    i -= 2;
                }
            }
        }
    }
}

```

```

        slen.erase(0);
        slen.append(s2, 1);

        lenth = slen.length();
        len = str.length();
    }
}

}

if (slen.compare("01") == 0) {
    str.erase(len - 1, 2);
    len = str.length();
}

cout << "\nУДАЛЕНЫ СЛОВА С 1 БУКВЫ: \n" << str << "\n";
}

//===== Функция подсчёта количества слов =====//
void count() {
    int slovo = 0, n = 0; // маркер наличия
                          слова и счётчик слов

    for (int i = 0; i < len; i++) {
        // если следующий элемент != (пробел, .?! ) и до этого слово не было
        // обнаружено, теперь слово обнаружено
        if ((str[i] != ' ' && slovo == 0) || (str[i] != ';' && slovo == 0) || (str[i] != '.' &&
slovo == 0) || (str[i] != '?' && slovo == 0) || (str[i] != '!' && slovo == 0)) {
            slovo = 1;
            n++;
        }
        // если след элемент = (пробел, .?!), слово закончилось
        else if ((str[i] == ' ') || (str[i] == ',') || (str[i] == '.') || (str[i] == '?') || (str[i] == '!')) {
            slovo = 0;
        }
    }

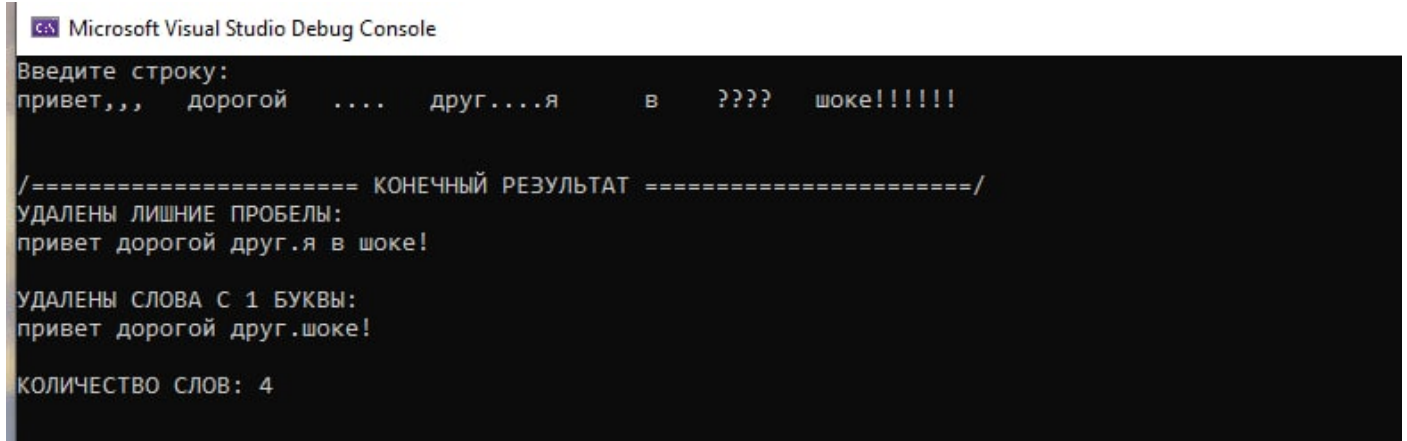
    cout << "\nКОЛИЧЕСТВО СЛОВ: " << n << "\n\n";
}

void main() {
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);

    create();
    del_space();
    del_w();
    count();
}

```

Результат:



```
Microsoft Visual Studio Debug Console
Введите строку:
привет,,, дорогой .... друг....я в ???? шоке!!!!!!

/===== КОНЕЧНЫЙ РЕЗУЛЬТАТ =====/
УДАЛЕНЫ ЛИШНИЕ ПРОБЕЛЫ:
привет дорогой друг.я в шоке!

УДАЛЕНЫ СЛОВА С 1 БУКВЫ:
привет дорогой друг.шоке!

КОЛИЧЕСТВО СЛОВ: 4
```

Рис. 1. Выведений результат виконання завдання 1

Висновки:

У ході виконання лабораторної роботи я дізналася про символічні рядки в мові програмування C++, їх особливості роботи та принцип застосування.

Я навчилася розробляти алгоритми та програми із застосуванням символічних рядків та використовувати їх для розв'язування за допомогою них поставлених математичних задач.