ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ

За курсом «Інформатика і Програмування»

Студента групи МС-19-1

Політіки Владислава Дмитровича

Лабораторна робота №9.2

**Тема**: «Обробка текстової інформації»

Варіант 11

**Завдання:** Написати програму, в якій були б розроблені 2 функції, що імітують дії стандартних рядкових функцій (див. індивідуальне завдання). Вибір функцій здійснюється за допомогою пунктів меню. У програмі продемонструвати роботу розробленої та стандартних функцій.

Індивідуальне завдання за варіантами:

**Варіант 11.** Стандартні функції strrchr( ) та strncmp( ).

**Про програму**

1. Опис розв’язку:

Для розробки меню я використав цикл з післяумовою такою, що при варіанті вибору виходу з програми, умова циклу не виконується (значення false). У програмі функцію перевірки вводу і сам ввід виконує функція PUT(). Вона перевіряє ввід за допомогою циклу з умовою “поки перший елемент масиву не 1, 2, 3 або 4 (можливі варіанти вибору) та його довжина не 1”, а при невірному вводі дає користувачу підказку. При вводі якогось варіанта, який був запропонований, оператор switch() здійснює перевірку того, яке число було введене. При тому чи іншому варіанті еквівалентності, здійснюється потрібний фрагмент коду (для цього був ще використаний оператор break). Далі розглянемо випадки, коли вибори варіантів еквівалентні числам 1, 2, 3 та 4.

При виборі варіанту 1 демонструється функція strrchr( ) та функція, що її імітує - imitator\_strrchr(). Ці функції шукають символ (тобто код символа), що був переданий функціям як другий параметр, у переданому масиві з кінця. Повертають ж ці функції вказівник на знайдений елемент або, коли символ у масиві не був знайдений, функції повертають нульовий вказівник, тобто вказівник, що ні на що не вказує. Опишемо функцію imitator\_strrchr(). Для початку вона шукає індекс нульового елемента у заданому масиві (цикл: поки елемент масиву з індексом ***i*** не є нульвим (\0), то ***i++*** (збільшення значення індексу на 1)), а далі розглядає всі індекси масиву, користуючись індексом останнього, починаючи з кінця - зі знайденого індекса нульового елемента. У випадку еквівалентності між заданою літерою (яка знаходиться у вигляді коду) та елементом масиву, що явно перетворюється у число типу беззвакового int, функція повертає вказівник на цей елемент - адресу пам’яті. Якщо еквівалентності не було знайдено, то функція повертає nullptr (нульовий вказівник).

При виборі варіанту 2 демонструється функція strncmp( ) та ідентична їй в плані результату - imitator\_strncmp. Ці функції мають три параметри. Перші два - вказівники на масиви, третій - параметр, що вказує кількість перших елементів у масивіі, що будуть перевірятись. Повертають функції значення або 1, або 0, або -1, в залежності від результату порівняння двох масивів. Значення 1 повертаеться, коли перша пара різних елементів-літер у різних масивах з однаковими індексами такі, що код у таблиці ASCII першого з них більший за код другого. Значення 0 - коли всі елементи масивів з однаковими індексами, що розглядаються, еквівалентні. Та значення -1 - коли перша пара різних елементів-літер у різних масивах з однаковими індексами такі, що код у таблиці ASCII другого з них більший за код першого. Розглянемо тепер функцію imitator\_strncmp(). Вона розглядає, як і було сказано, всі елементи, що потрібно перевірити - третій параметр функції. Це робить цикл for. В циклі перевіряються пари відповідних елементів. Якщо елементи однакові, то цикл продовжується до тих пір доки не трапиться останній заданий елемент, що потрібно перевірити, чи признак кінця в обох масивах одночасно (нульовий елемент \0), чи доти, доки не трапляться різні елементи. В останньому випадку у функції будуть перевірені нерівності між кодами символів (у програмі коди символів дістаються шляхом явного перетворення типу char у тип unsigned int) та у випадку виконання тої чи іншої нерівності повертатися відповідно значення 1 або -1. У випадках, коли цикл дійшов до індексу, що елементи масивів з цим індексом рівні та еквівалентні нульвому символу \0, або коли всі символи були перевірені, та не виконувались вище розглянуті нерівності, то тоді повертається число 0.

Тож описи дій моїх функції дійсно такі, що їх результат еквівалентний результатам стандартних функцій.

При виборі варіанту 3 очищається консоль та виводяться деякі дані, що потрібно залишити (наприклад: Лабораторная работа 9.2", " Вариант 11", а при наступній ітерації циклу видається список можливих варіантів продовження цієї ітерації знову).

При виборі варіанту 4 основний цикл для меню отримує значення false і програма закінчує своє виконання.

1. Вихідний текст програми розв’язку задачі:

#include <iostream>

#include <Windows.h>

using namespace std;

void PUT(char var[])

{

cout << " Введите вариант: ";

cin.getline(var, 100);

while ((var[0] != '1' && var[0] != '2' && var[0] != '3' && var[0] != '4') || (strlen(var) != 1))

{

cout << " Внимание! Введённое число должно быть единицей или двойкой!" << endl;

cout << " Введите вариант: ";

cin.getline(var, 100);

}

}

char\* imitator\_strrchr(char\* str, int symbol)

{

int last\_symbol;

for (last\_symbol = 0; str[last\_symbol] != '\0'; last\_symbol++);

while (last\_symbol >= 0)

{

if ((int)str[last\_symbol] == symbol)

return &str[last\_symbol];

last\_symbol--;

}

return nullptr; // нулевой указатель

}

int imitator\_strncmp(const char\* str1, const char\* str2, int len\_act)

{

for (int i = 0; i < len\_act; i++)

{

if(str1[i] == '\0' && str2[i] == '\0') return 0;

if((unsigned int)str1[i] > (unsigned int)str2[i]) return 1;

if((unsigned int)str1[i] < (unsigned int)str2[i]) return -1;

}

return 0;

}

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

char var[100], string[500],additional\_string[500], symbol; int universal;

cout << " Лабораторная работа 9.2" << endl;

cout << " Вариант 11" << endl;

do

{

cout << " Выберете одну из стандартных функций:" << endl;

cout << " 1) функция strrchr()\n 2) функция strncmp()\n 3) очистить консоль\n 4) выход с программы\n";

PUT(var);

switch (var[0])

{

case '1':

cout << " Функция strrchr()" << endl;

cout << " Введите строку: "; cin.getline(string, 500);

cout << " Введите символ для переобразования его в кодировку таблицы ASCII: ";

cin >> symbol;

while (cin.get() != '\n')

{

while (cin.get() != '\n');

cout << " Вы должны ввести один символ!" << endl;

cout << " Введите символ для переобразования его в кодировку таблицы ASCII: ";

cin >> symbol;

}

universal = (unsigned int)symbol;

cout << " Код введённого символа: " << universal << endl << endl;

cout << " Результат моей функции (значение - указатель на последний элемент, соотвецтвующий введённому): ";

cout << imitator\_strrchr(string, universal);

cout << endl << " Результат стандартной функции (значение такое же): ";

cout << strrchr(string, universal);

cout << endl << endl;

break;

case '2':

cout << " Функция strncmp()" << endl;

cout << " Введите строку 1: "; cin.getline(string, 500);

cout << " Введите строку 2: "; cin.getline(additional\_string, 500);

cout << " Введите число n, которое обозначает первые n символов введённых массивов: ";

cin >> universal;

cout << endl;

cout << " Результат моей функции: ";

cout << imitator\_strncmp(string, additional\_string, universal);

cout << endl << " Результат стандартной функции: ";

cout << strncmp(string, additional\_string, universal);

cout << endl << endl;

break;

case '3': system("CLS");

cout << " Лабораторная работа 9.2" << endl;

cout << " Вариант 11" << endl;

}

}

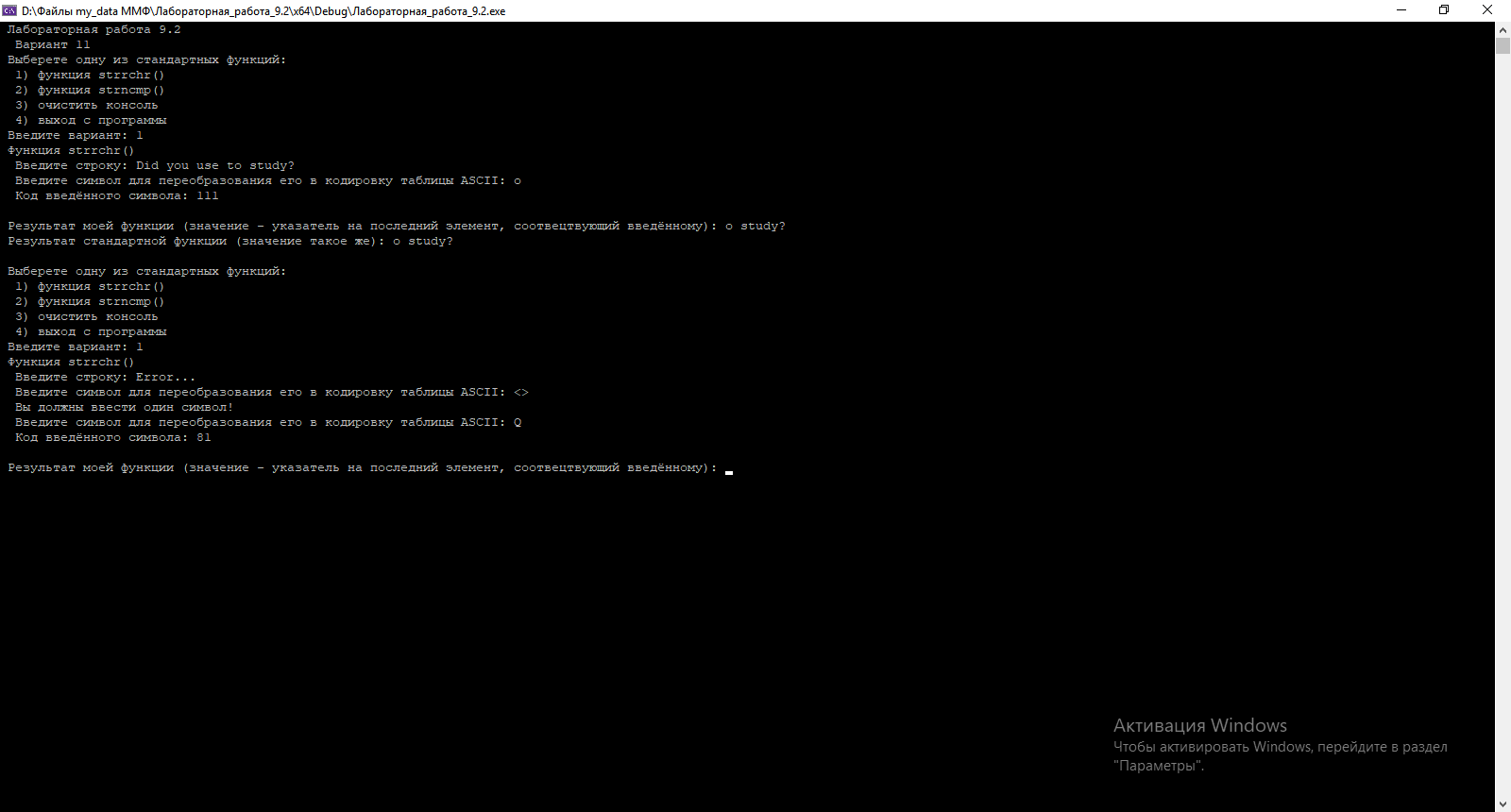
while (var[0] != '4');

return 0;

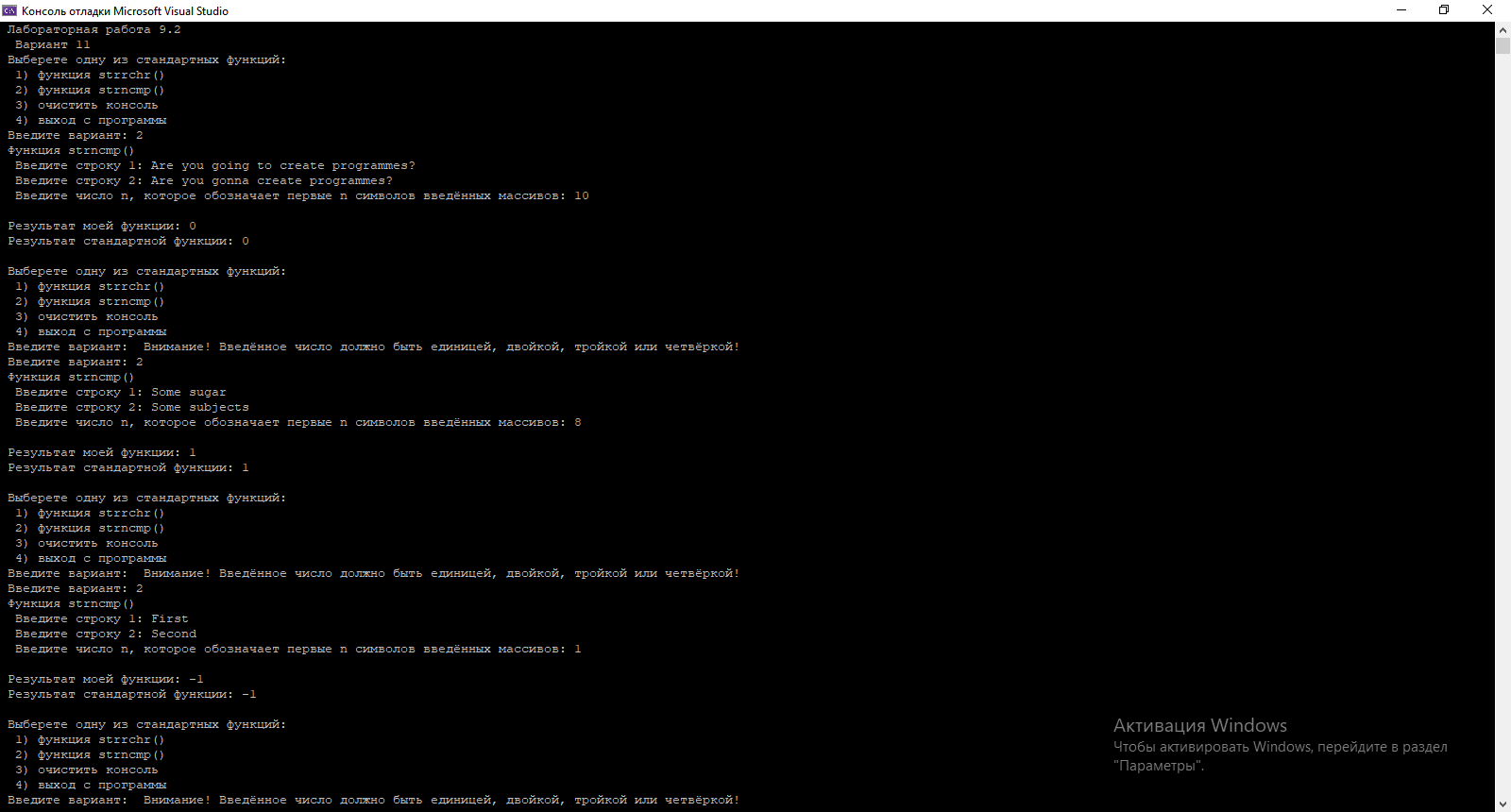
}

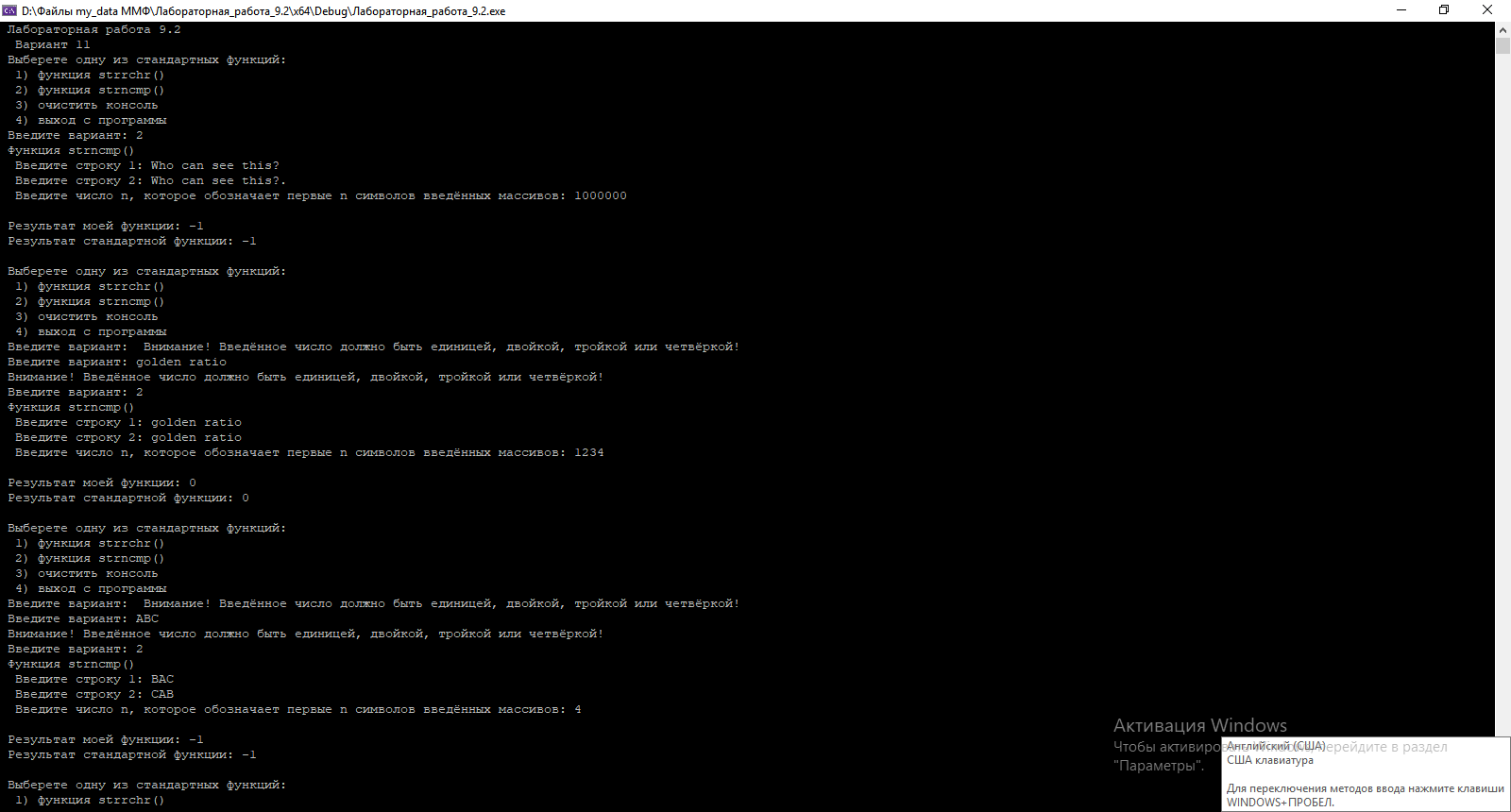
1. Опис інтерфейсу:

Показано роботу однієї з функцій(функція strrchr( )). Тут оператор cout виводить частину введеного речення, починаючи з введеної літери, яка є останньою у реченні (літера явно перетворюється в її код у таблиці ASCII, далі цей код передається у функції). Але у випадку, коли літери не було знайдено, то передається нульовий вказівник - вивід здійснити не вдається.



Інтерфейс програми (функція strncmp( ))





1. Опис тестових прикладів:

При всіх моїх тестових прикладах, програма видає правильні результати.

1. Аналіз помилок:

Помилок не має.