Министерство образования и науки Российской Федерации   
Федеральное государственное бюджетное образовательное   
учреждение высшего образования   
«Алтайский государственный технический университет им. И.И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчёт защищён с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Астахова Е.В.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г.

Отчёт

по лабораторной работе № 3

«Форматы данных в ЭВМ»

по дисциплине «Основы программной инженерии»

Студенты группы ПИ 92:

В.М. Шульпов, И.В. Вильк, А. Н. Гулин, Ю.П. Пирязев

Преподаватель доцент, к. п. н. Астахова Е.В.

Барнаул 2019

**Задание 1** Посимвольная обработка строк

* В исходной строке удалить лишние пробелы.

**Код программы**

//ЛАБОРАТОРНАЯ РАБОТА №3

//ЗАДАНИЕ 1

//Программа удаляет лишние пробелы в исходной строке

#include "stdafx.h"

#include <locale.h>

#include <string.h>

#include <Windows.h>

int main(int argc, char \*argv[]) {

setlocale(LC\_ALL, "Rus");

system("color F0");

char str[101] = "";

char res[101] = "";

printf("Введите строку:");

fgets (str, 101, stdin);

int j = 0;

int i;

for (i = 0; i<strlen(str) ; i++) { //Цикл работает в зависимости от длины строки

if (str[ i] == ' ') //Если встречается пробел

{

if (j==0) continue; //Если пробел вначале, то он пропускается при вводе в конечную строку

if (str[ i+1] == ' ') continue; //Если 2 пробела, то один из них пропускается

}

res[j] = str[ i]; //Ввод нужных символов в новый массив

j++;

}

i=strlen(res);

if(res[ i-2] == ' ') //Проверяет есть ли пробелы в конце

res[ i-2] = '\0'; //Убирает пробелы

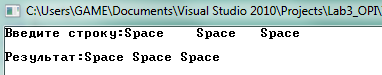
printf("\nРезультат:%s", res);

getchar();

return 0;

}

**Тест программы**



**Задание 2** Пословная обработка строк

* В исходной строке найти два одинаковых слова длиной более 5 символов, скопировать их в выходную строку и удалить из исходной строки. Повторять этот процесс, пока такие фрагменты находятся. Остаток строки добавить в выходную строку. В выходной строке не должно быть дубликатов слов.

**Код программы**

//ЛАБОРАТОРНАЯ РАБОТА №3

//ЗАДАНИЕ 2

//Пословная обработка строк

#include "stdafx.h"

#include <locale.h>

#include <string.h>

#include <Windows.h>

#include <conio.h>

int main(int argc, char \*argv[])

{

setlocale(LC\_ALL, "Rus");

system("color F0");

char input[100]; // объявление входной строки

char output[100] = { 0 }; // объявление выходной строки

char f = 1; // объявление флага

printf("Введите строку с одинаковыми словами: ");

gets\_s(input);

printf("Деление строки на слова:\n");

while (f) { // пока флаг не равен нулю

f = 0;

memset(output, 0, 100 \* sizeof(char)); // обнуление выходной строки

int tokensCount = 0; // объявление счётчика слов

char\* tokens[20]; // объявление массива указателей слов

char\* token\_next = NULL; // обращение в ноль вспомогательного токена для strtok\_s

char\* token = strtok\_s(input, " ,.-", &token\_next); // присваивание токену первого слова строки (до разделителя)

while (token != NULL) //пока есть необработанные слова

{

tokens[tokensCount++] = token; //tokens[i] = token

token = strtok\_s(NULL, " ,.-", &token\_next); // token++

}

for (int i=0; i < tokensCount; i++) // первое слово

{

if (strlen(tokens[i]) <= 5) continue; // проверка на условие о длине повторяющихся слов

for (int j = i + 1; j < tokensCount; j++) // второе слово

{

if (strlen(tokens[j]) <= 5) continue; // проверка на условие о длине повторяющихся слов

if (strcmp(tokens[i], tokens[j] == 0)) // сравнение слов

{

f = 1; // установка флага в единицу, дошли до одинаковых слов

strcat(output, tokens[i]); // выходная строка (изначально ноль + i-ый токен)

strcat(output, " "); // добавление пробела

for (int m = i; m < tokensCount; m++) tokens[m] = tokens[m + 1]; // удаление

for (int m = j - 1; m < tokensCount; m++) tokens[m] = tokens[m+1]; // удаление

tokensCount -= 2; // вычитание из-за удаления повторяющихся слов

}

}

}

for (int i = 0; i < tokensCount; i++) {

strcat(output, tokens[i]); // добавление неповторяющихся слов

strcat(output, " "); // добавление пробела

}

strcpy(input, output); // копирование выходной строки во входную

}

puts(output);

\_getch();

return 0;

}

**Задание 3** Вычисления

* Вычислить площадь треугольника.

**Код программы**

// exercise3.cpp

//ЛАБОРАТОРНАЯ РАБОТА №3

//ЗАДАНИЕ 3

//Программа вычисляет площадь треугольника

#include "stdafx.h"

#include <stdio.h>

#include <locale.h>

#include <Windows.h>

int \_tmain(int argc, \_TCHAR\* argv[])

{

int a, b; // a - основание треугольника, b - высота треугольника

float s; // s - искомая площадь

system("color F0");

setlocale(LC\_ALL, "Rus");

do

{

printf("Введите основание и высоту треугольника (целые), не превосходящие 1000!\n");

scanf("%d%d", &a, &b);

} while (1 > a || a > 1000 || 1 > b || b > 1000);

s = a\*b / 2;

printf("Площадь треугольника = %.1f\n", s);

return 0;

}

**Тест программы**

