### 

КОМИТЕТ ПО ОБРАЗОВАНИЮ САНКТ-ПЕТЕРБУРГА

**САНКТ-ПЕТЕРБУРГСКОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**«КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»**

**ОТЧЕТ**

**Основы алгоритмизации и программирования**

**Практическая работа 11. Модули**

Работу выполнил: Танасов Евгений Эдуардович

Группа:324 Специальность: 09.02.07

Преподаватель: Смирнова Ирина Петровна

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«11» декабря 2024 г.

Преподаватель: Смирнова Ирина Петровна

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_» сентября 2021 г.

Санкт-Петербург 2021

Санкт-Петербург 2015 г.

Оглавление

[**Цель практической работы:** 3](#_Toc181811567)

[**Исходный текст программы:** 3](#_Toc181811568)

[Результат работы программы**:** 4](#_Toc181811570)

# **Цель практической работы:**

Познакомиться с основами модульного программирования

# **Исходный текст программы:**

**Основной модуль main.c**

#include <stdio.h>

#include <locale.h>

#include "word\_directive.h"

int main() {

setlocale(LC\_ALL, "rus");

char inputF[] = "input.txt";

char outputF[] = "output.csv";

Words words[MAX\_WORDS];

int count = 0;

sortWords(words, count);

printf("Результат находится в файле %s, проверьте корень проекта\n", outputF);

}

**Модуль interaction\_word.c**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include "word\_directive.h"

void addWord(Words words[], int\* count, const char\* word) {

for (int i = 0; i < \*count; i++) {

if (strcmp(words[i].word, word) == 0) {

words[i].rate++;

return;

}

}

if (\*count < MAX\_WORDS) {

strcpy(words[\*count].word, word);

words[\*count].rate = 1;

(\*count)++;

}

}

int compareWords(const void\* a, const void\* b) {

Words\* wordA = (Words\*)a;

Words\* wordB = (Words\*)b;

return strcmp(wordA->word, wordB->word);

}

void sortWords(Words words[], int count) {

for (int i = 0; i < count - 1; i++) {

for (int j = 0; j < count - i - 1; j++) {

if (strcmp(words[j].word, words[j + 1].word) > 0) {

Words temp = words[j];

words[j] = words[j + 1];

words[j + 1] = temp;

}

}

}

}

int readWords(Words words[], int\* count, const char\* filename) {

FILE\* file = fopen(filename, "r");

if (!file) {

return 0;

}

char temp[MAX\_LENGTH];

while (fscanf(file, "%s", temp) != EOF) {

addWord(words, count, temp);

}

fclose(file);

return 1;

}

int writeWords(Words words[], int count, const char\* filename) {

FILE\* file = fopen(filename, "w");

if (!file) {

return 0;

}

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

fprintf(file, "%s - %d шт.\n", words[i].word, words[i].rate);

}

fclose(file);

return 1;

}

**Заголовочный файл word\_directive.h**

#ifndef WORD\_DIRECTIVE\_H

#define WORD\_DIRECTIVE\_H

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <locale.h>

#define MAX\_LENGTH 100

#define MAX\_WORDS 1000

typedef struct {

char word[MAX\_LENGTH];

int rate;

} Words;

void addWord(Words words[], int\* count, const char\* word);

void sortWords(Words words[], int count);

int readWords(Words words[], int\* count, const char\* filename);

int writeWords(Words words[], int count, const char\* filename);

#endif

Результат работы программы**:**

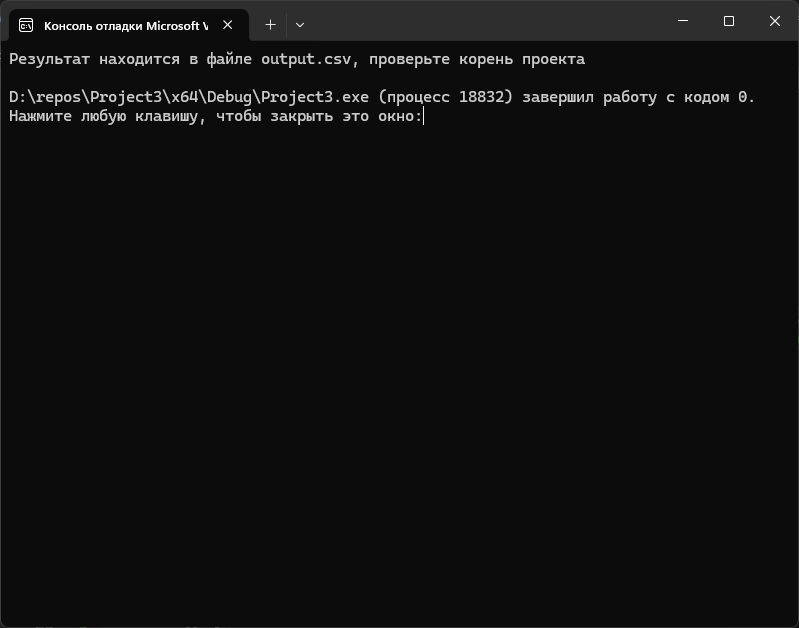


Рисунок 1 – Тест 1

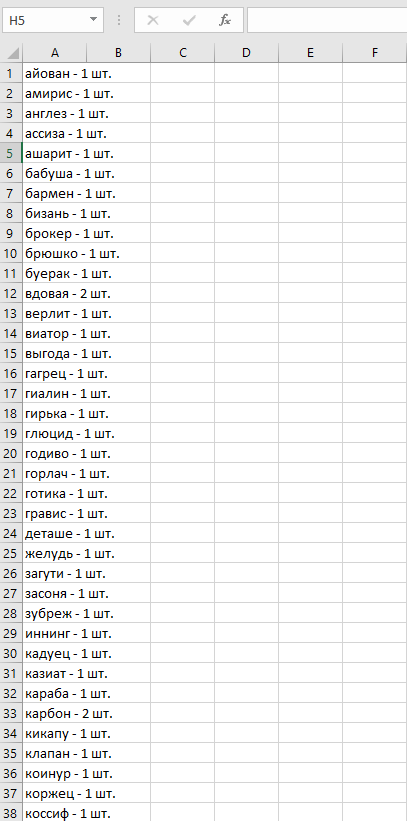


Рисунок 2 – Проверка в Excel №1

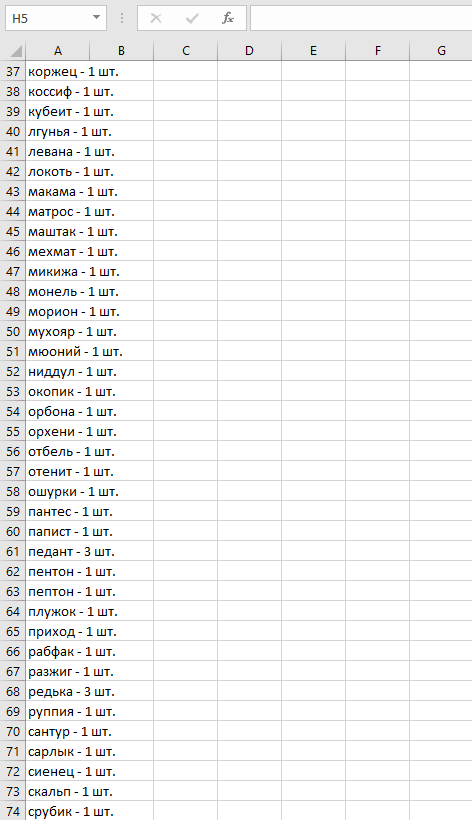


Рисунок 2 – Проверка в Excel №2

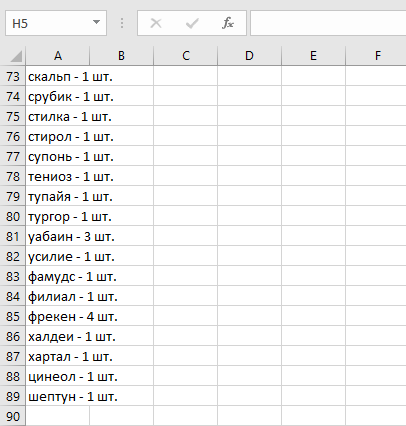


Рисунок 3 – Проверка в Excel №3

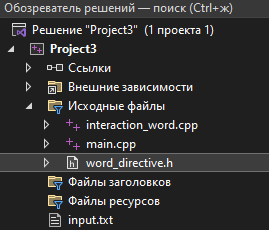


Рисунок 4 – Структура файлов проекта