Итоговая работа по курсу «Программирование на языке С (базовый уровень)»

Мухаметшин Роман Винерович

Оглавление

[Постановка задания 3](#_heading=h.3znysh7)

[Описание программы 5](#_heading=h.2et92p0)

[Исходный код программы 5](#_heading=h.tyjcwt)

[Описание файлов программы 5](#_heading=h.3dy6vkm)

[Описание работы программы 6](#_heading=h.1t3h5sf)

[Примеры написания командной строки для запуска: 6](#_heading=h.4d34og8)

[Демонстрация работы программы 7](#_heading=h.2s8eyo1)

[Сборка программы утилитой make 8](#_heading=h.17dp8vu)

# Постановка задания

**Создание консольного приложения — Статистика температуры**

Необходимо реализовать консольное приложение, которое осуществляет считывание текстового файла csv, состоящего из строк следующего формата:

**YEAR;MONTH;DAY;HOUR;MINUTE;TEMPERATURE**

dddd;mm;dd;hh;mm;temperature

dddd - год 4 цифры

mm - месяц 2 цифры

dd - день 2 цифры

hh - часы 2 цифры

mm - минуты 2 цифры

temperature - целое число от -99 до 99

В архиве файле хранится статистика собранная датчиком температуры за 1 календарный год. Предполагается, что датчик собирал информацию не чаще чем 1 раз в минуту и сохранял в заданном формате каждое значение в текстовый файл с новой строки.

В какой-то момент времени датчик мог не работать, тогда данные поэтому периоду могут отсутствовать. Пример входного файла:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| YEAR | MONTH | DAY | HOUR | MINUTE | TEMPERATURE |
| 2021 | 1 | 1 | 23 | 1 | -5 |
| 2021 | 1 | 1 | 23 | 3 | -6 |
| 2021 | 1 | 1 | 23 | 1 | -7 |
| 2021 | 1 | 2 | 7 | 5 | -10 |

**Требования к обработке данных**

Необходимо вывести статистику по каждому месяцу, с учетом исходных данных:

* среднемесячная температура
* минимальная температура в текущем месяце
* максимальная температура в текущем месяце

Также необходимо вывести статистику за год:

* среднегодовая температура
* минимальная температура
* максимальная температура

**Требования к аргументам командной строки**

Приложение должно обрабатывать аргументы командной строки:

минимальный набор поддерживаемых ключей:

* -h Описание функционала приложения. Список ключей, которые обрабатывает данное приложение и их назначение.
* -f <filename.csv> входной файл csv для обработки.
* -m <номер месяца> если задан данный ключ, то выводится только статистика за указанный месяц.
* если нет параметров, то выдается help

**Требования к ошибкам в входных данных**

* Приложение должно корректно работать на любых входных данных, если формат csv файла не соответствует заданному, то необходимо указать номер строки файла csv, в которой обнаружена ошибка и не учитывать данную строку.

В архиве с заданием [temperature\_data\_examples.zip](https://drive.google.com/file/d/1BBWwFSzVtGRPgvout8EmWXsdVUEVRtYG/view?usp=drive_link) лежат два файла:

* temperature\_big.csv — файл со статистикой за год
* temperature\_small.csv — укороченный файл с ошибками для тестирования

**Требования к сборке приложения**

* Приложение должно собираться при помощи утилиты make.
* Все прототипы функций, используемые в приложении, должны быть вынесены в отдельный файл temp\_functions.h
* Тексты функций — в файл temp\_functions.c
* Для реализации приложения рекомендуется использовать массив из структурного типа данных для хранения показаний датчика.

# Описание программы

## Исходный код программы

|  |
| --- |
| * Ссылка на репозиторий: * ***https://github.com/AVR42/MFTI-IoT/tree/7eb9a888c22cd6f9715173d2a5ec7019be61aeff/HW13*** |

## Описание файлов программы

Программа состоит из следующих файлов:

**main.c** – основной файл программы с точкой входа, функцией main. Обрабатывает аргументы командной строки (-f, -m, -h), запускает чтение CSV-файла , в зависимости от параметров: показывает статистику за конкретный месяц (-m <номер>) или статистику по всем месяцам + годовую.

**temp\_function.h** – файл прототипов функций программы.

**temp\_function.c** – файл описания функций программы. Содержит функции:

* void print\_help() - Показывает справку по ключам запуска.
* int read\_csv\_file(const char \*filename, Reading \*\*readings) - Читает CSV-файл, заполняет массив.
* void print\_month\_stats(const Reading \*readings, int count, int month) - Печатает статистику (среднюю, минимальную и максимальную температуру) за один конкретный месяц.
* void print\_all\_months\_stats(const Reading \*readings, int count) — печатает статистику по каждому месяцу.
* void print\_year\_stats(Reading \*readings, int count) - Собирает и выводи таблицу по месяцам: средняя, мин., макс общую среднегодовую температуру, общий минимум и максимум.

**makefile** – файл инструкция для сборки;

**temp\_stats**.exe – собранный исполняемый файл программы.

## Описание работы программы

***Пример оформления блока:***

Программа является консольным приложением и рекомендуется запускать её из командной строки.

При запуске без указания ключей, программа выведет информацию как с ключем «-h» для ознакомления с возможностями программы..

Программа допускает применение следующих ключей:

* «-h» - получение информации о возможных ключах запуска с кратким описанием их назначения;
* «-f file\_name» - указание файла для обработки, где file\_name – имя файла. Если в этом режиме не добавлена опция «-m», то выведется полная статистика по всему файлу;
* «-m xx» - указание месяца для получения статистики по конкретному месяцу, где xx - месяц;

### Примеры написания командной строки для запуска:

«./temp\_stats»

«./temp\_stats -h»

«./temp\_stats -f temperature\_small.csv»

«./temp\_stats -f temperature\_big.csv -m 3»

«./temp\_stats -f temperature\_big.csv -m 3»

### Демонстрация работы программы

|  |
| --- |
| Рисунок 1. Запуск программы с опцией «-h» |
| Рисунок 2. Запуск программы с указанием файла и опцией выбора месяца    Рисунок 3. Запуск программы с указанием имени файла |

### Сборка программы утилитой make

|  |
| --- |
|  |
|  |