



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

Михайличенко Дмитрий Леонидович

# Оглавление

Постановка задания	3
Описание программы	5
Исходный код программы	5
Описание файлов программы	5
Описание работы программы	6
Примеры написания командной строки для запуска:	6
Демонстрация работы программы	7
Сборка программы утилитой make	8

# 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 архивном 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 по этому 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](#) 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

[https://github.com/hhrshh/MFTI\\_C/tree/main/coursework](https://github.com/hhrshh/MFTI_C/tree/main/coursework)

## Описание файлов программы

Программа состоит из следующих файлов:

- ✦ main.c – основной файл программы с функцией main. Содержит только логику запуска функций и обработку ключей, поступающих от командной строки;
- ✦ main.h – содержит макросы описания ключей командной строки
- ✦ temp.h – файл прототипов функций программы;
- ✦ temp.c – файл описания функций программы;
- ✦ Makefile – файл инструкция для утилиты сборки make;

## Описание работы программы

- ❖ Программа является консольным приложением и рекомендуется запускать её из командной строки Linux. При запуске без указания ключей, программа кратко выведет информацию о своем назначении и предложении ввести ключ «-h» для получения инструкций. Программа допускает применение следующих ключей:
- ❖ -h - получение информации о возможных ключах запуска с кратким описанием их назначения;
- ❖ -f - 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 - указание месяца для получения статистики по конкретному месяцу, где х - месяц;
- ❖ Так же в случае успешного прочтения файла команда выведет количество прочтенных строк и строки с ошибкой

Примеры написания командной строки для запуска:

```
./temp
```

```
./temp -h
```

```
./temp -f small_file.csv
```

```
./temp -f big_file.csv -m 3
```

## Демонстрация работы программы

Снимки экрана запуска программы:

Рисунок 1. Запуск программы с опцией «-h»

```
hrsh@hrsh-PC:~/mfti_dz/coursework$ ./temp -h
Usage: temp [OPTION]... [FILE_PATH]
  -h display this help text.
  -f the path to the CSV file containing temperature data.
  -m <month_number> if this option is specified, displayed only statistics for the specified month.
Description: a console application for reading temperature values from a CSV file.
hrsh@hrsh-PC:~/mfti_dz/coursework$
```

Рисунок 2. Запуск программы с указанием файла и опцией выбора месяца

```
hrsh@hrsh-PC:~/mfti_dz/coursework$ ./temp -f temperature_small.csv -m 5
=====
Number of lines read with error: 4 10 11 12
Total values read: 12
=====
# Year Month NuValue ErValue MonthAvg MonthMax MonthMin
5 2021 5 1 0 10 10 10
=====
hrsh@hrsh-PC:~/mfti_dz/coursework$
```

Рисунок 3. Запуск программы с указанием имени файла

```
hrsh@hrsh-PC:~/mfti_dz/coursework$ ./temp -f temperature_small.csv
=====
Number of lines read with error: 4 10 11 12
Total values read: 12
=====
# Year Month NuValue ErValue MonthAvg MonthMax MonthMin
1 2021 1 3 4 -44 -43 -47
2 2021 2 2 0 -27 -25 -30
3 2021 3 1 0 -10 -10 -10
4 2021 4 1 0 0 0 0
5 2021 5 1 0 10 10 10
6 2021 6 1 0 25 25 25
7 2021 10 1 0 2 2 2
8 2021 11 1 0 -5 -5 -5
9 2021 12 1 0 -20 -20 -20
=====
Statistic for year: avarage t = -15, max t = 25, min t = -47
=====
hrsh@hrsh-PC:~/mfti_dz/coursework$
```

## Сборка программы утилитой make

Снимки экрана запуска программы:

Рисунок 1. Запуск программы make

```
hrsh@hrsh-PC:~/mfti_dz/coursework$ make
gcc -c -o main.o main.c
gcc -c -o temp.o temp.c
gcc -o temp main.o temp.o
hrsh@hrsh-PC:~/mfti_dz/coursework$
```

Рисунок 2. Запуск программы make опцией clean

```
hrsh@hrsh-PC:~/mfti_dz/coursework$ make clean
rm -rf *.o temp
hrsh@hrsh-PC:~/mfti_dz/coursework$
```