



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

**Дубов Денис Александрович**

# 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 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 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/fwdden/embedded\\_c/tree/main/12\\_final](https://github.com/fwdden/embedded_c/tree/main/12_final)

## Описание файлов программы

Программа состоит из следующих файлов:

main.c – основной файл программы с точкой входа, функцией main. Содержит только логику запуска функций и обработку ключей, поступающих от командной строки;

temp\_api.h – файл прототипов функций программы;

temp\_api.c – файл описания функций программы;

makefile – файл инструкция для утилиты сборки mingw32-make;

sensor.exe – собранный исполняемый файл программы.

## Описание работы программы

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

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 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;

- «-t» - сортировка собранной статистики по убыванию температуры. Требуется использовать вместе с ключом «-f file\_name»;
- «-d» - сортировка собранной статистики по дням. Требуется использовать вместе с ключом «-f file\_name»;

**Примеры написания командной строки для запуска:**

`sensor_app`

`sensor_app -h`

`sensor_app -f temperature_small.csv`

`sensor_app -f temperature_small.csv -t`

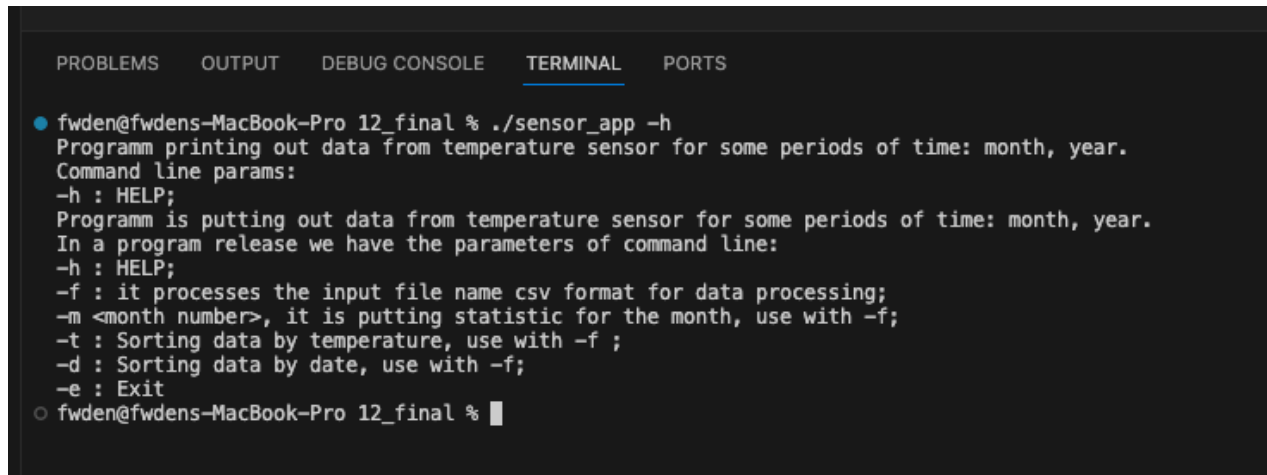
`sensor_app -f temperature_small.csv -d`

`sensor_app -f temperature_big.csv -m 3`

## Демонстрация работы программ

### Образцы снимков:

Рисунок 1. Запуск программы с опцией «-h»



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● fwden@fwdens-MacBook-Pro 12_final % ./sensor_app -h
Programm printing out data from temperature sensor for some periods of time: month, year.
Command line params:
-h : HELP;
Programm is putting out data from temperature sensor for some periods of time: month, year.
In a program release we have the parameters of command line:
-h : HELP;
-f : it processes the input file name csv format for data processing;
-m <month number>, it is putting statistic for the month, use with -f;
-t : Sorting data by temperature, use with -f ;
-d : Sorting data by date, use with -f;
-e : Exit
○ fwden@fwdens-MacBook-Pro 12_final %
```

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



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● fwden@fwdens-MacBook-Pro 12_final % ./sensor_app -f temperature_big.csv -m 4
Programm printing out data from temperature sensor for some periods of time: month, year.
Command line params:
-h : HELP;

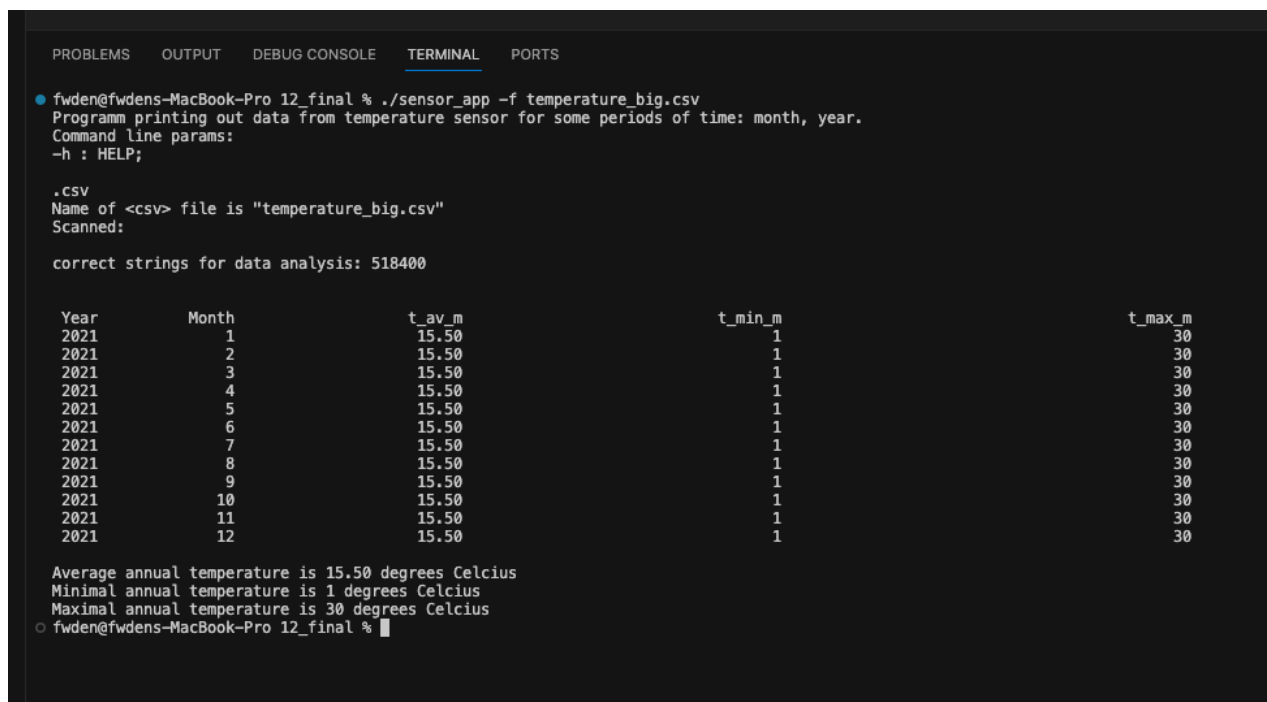
.csv
Name of <csv> file is "temperature_big.csv"
Scanned:

correct strings for data analysis: 518400

Number of month is 4

Year      Month      t_av_m      t_min_m      t_max_m
2021      4          15.50       1            30
○ fwden@fwdens-MacBook-Pro 12_final %
```

Рисунок 3. Запуск программы с указанием имени файла



The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

fwden@fwdens-MacBook-Pro 12\_final % ./sensor\_app -f temperature\_big.csv  
Programm printing out data from temperature sensor for some periods of time: month, year.  
Command line params:  
-h : HELP;

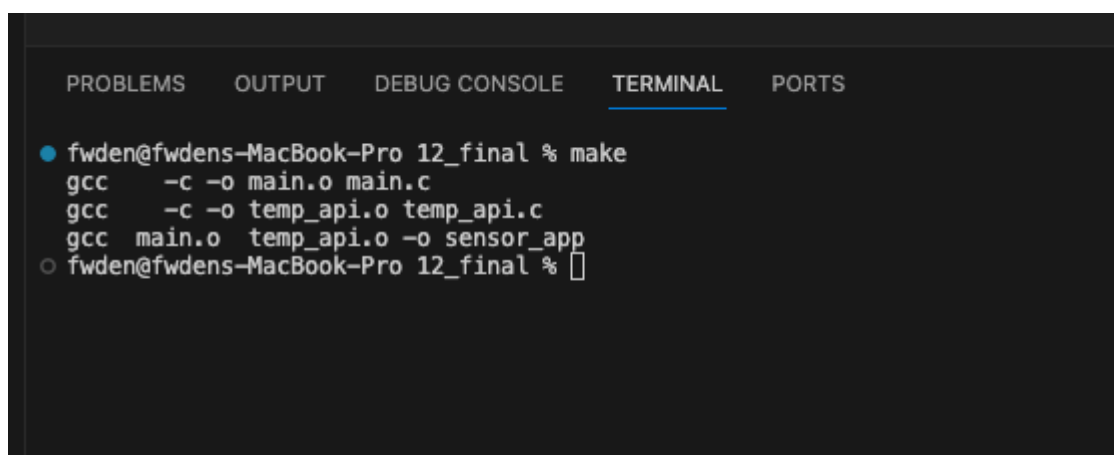
.csv  
Name of <csv> file is "temperature\_big.csv"  
Scanned:  
  
correct strings for data analysis: 518400

Year	Month	t_av_m	t_min_m	t_max_m
2021	1	15.50	1	30
2021	2	15.50	1	30
2021	3	15.50	1	30
2021	4	15.50	1	30
2021	5	15.50	1	30
2021	6	15.50	1	30
2021	7	15.50	1	30
2021	8	15.50	1	30
2021	9	15.50	1	30
2021	10	15.50	1	30
2021	11	15.50	1	30
2021	12	15.50	1	30

Average annual temperature is 15.50 degrees Celcius  
Minimal annual temperature is 1 degrees Celcius  
Maximal annual temperature is 30 degrees Celcius  
fwden@fwdens-MacBook-Pro 12\_final %

## Сборка программы утилитой make

Снимок сборки программы утилитой make.



The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

fwden@fwdens-MacBook-Pro 12\_final % make  
gcc -c -o main.o main.c  
gcc -c -o temp\_api.o temp\_api.c  
gcc main.o temp\_api.o -o sensor\_app  
fwden@fwdens-MacBook-Pro 12\_final %