



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

**Зинкина Дмитрия Игоревича**

# Оглавление

Постановка задания

Описание программы

- Исходный код программы

- Описание файлов программы

- Описание работы программы

  - Примеры написания командной строки для запуска:

  - Демонстрация работы программы

  - Сборка программы утилитой `make`

# 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/StepD32/Home\\_task/tree/main/Final\\_task](https://github.com/StepD32/Home_task/tree/main/Final_task)

## 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, а также проверяет корректность открытия файла на чтение и закрытие его;

temp\_function.h – файл прототипов функций программы;

temp\_function.c – файл описания функций программы;

makefile – файл инструкция для утилиты сборки mingw32-make;

prog.exe – собранный исполняемый файл программы.

## Описание работы программы

Программа является консольным приложением и запускается с помощью командной строки.

При запуске без указания ключей, программа кратко выведет информацию о своем назначении и предложении ввести ключ «-h» для получения дополнительных ключей .

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;  
Пример: prog.exe -f temperature\_big.csv
- «-m xx» - указание месяца для получения статистики по конкретному месяцу, где xx - месяц;  
Пример: prog.exe -f temperature\_big.csv -m 7

Примеры написания командной строки для запуска:

prog.exe

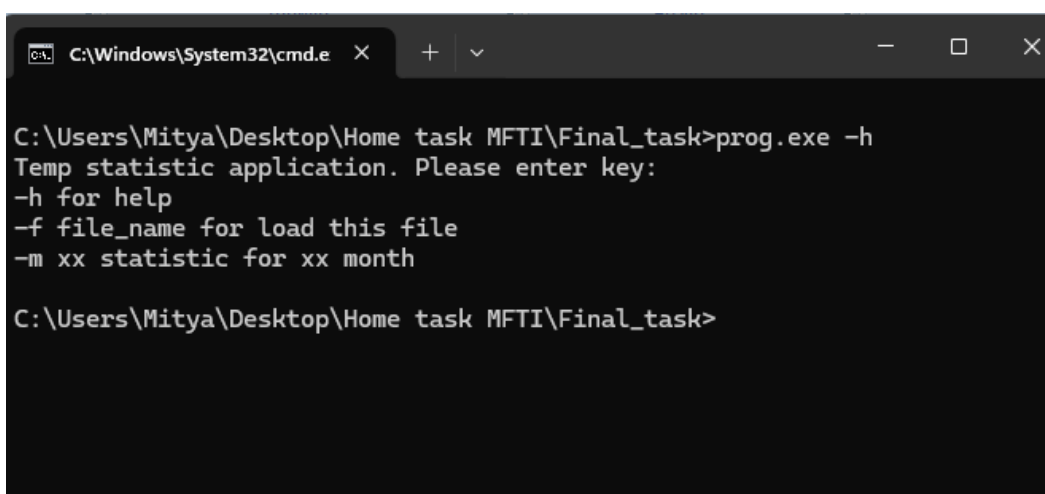
prog.exe -h

prog.exe -f temperature\_small.csv

prog.exe -f temperature\_big.csv -m 3

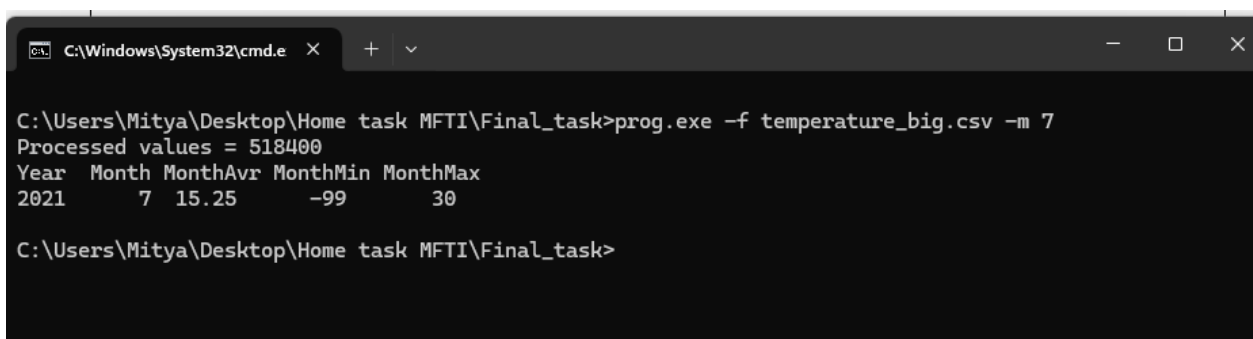
## Демонстрация работы программы

Рисунок 1. Запуск программы с опцией «-h»



```
C:\Windows\System32\cmd.e X + v
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>prog.exe -h
Temp statistic application. Please enter key:
-h for help
-f file_name for load this file
-m xx statistic for xx month
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>
```

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



```
C:\Windows\System32\cmd.e X + v
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>prog.exe -f temperature_big.csv -m 7
Processed values = 518400
Year Month MonthAvr MonthMin MonthMax
2021 7 15.25 -99 30
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>
```

Рисунок 3. Запуск программы с указанием имени файла

```
C:\Windows\System32\cmd.e X + v
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>prog.exe -f temperature_big.csv
Processed values = 518400
Year Month MonthAvr MonthMin MonthMax
2021 1 15.65 1 99
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 15.25 -99 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 15.43 -99 30
Year statistic: average 15.48647, min -99, max 99
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>
```

## Сборка программы утилитой make

```
C:\Windows\System32\cmd.e X + v
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>mingw32-make clean
del /f *.o prog.exe

C:\Users\Mitya\Desktop\Home task MFTI\Final_task>mingw32-make all
gcc -g -std=c99 -Wall -c -o main.o main.c
main.c: In function 'main':
main.c:71:2: warning: implicit declaration of function 'static_output_node'; did you mean 'static_output'? [-Wimplicit-function-declaration]
   71 |     static_output_node(pHead, &Month);
      |     ^~~~~~~~~~~~~~~~~~
      |     static_output
gcc -g -std=c99 -Wall -c -o temp_functions.o temp_functions.c
gcc -g -std=c99 -Wall -o prog main.o temp_functions.o
C:\Users\Mitya\Desktop\Home task MFTI\Final_task>
```

e