



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

Павлов Е.И.

Москва, 2026 г.

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	8
Сборка программы утилитой make	10
Потенциал развития программы	10

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 период времени, который может включать один 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	4	-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

Исходный код программы доступен в репозитории:

https://github.com/YogeneP/mipt_hw_C/tree/a780e9b56fda5cb39e7dc17940571aece d383c8e/HW_TEMP_PRJ

Описание файлов программы

Программа состоит из следующих файлов:

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

temp_api.h – файл прототипов функций программы;

temp_api.c – файл описания функций программы;

makefile – файл инструкция для утилиты сборки mingw32-make / make; сконфигурирован для автоматического определения семейства ОС и сборки программы для ОС Windows или Linux

run_temp.exe – собранный исполняемый файл программы для ОС Windows.

Описание работы программы

Для реализации расширенного функционала и обеспечения возможности дальнейшего развития при разработке программы, кроме требований исходного технического задания, были реализованы следующие возможности:

- Поддержка файлов данных за несколько лет.
- При чтении файла данных выводятся на печать номера и содержимое строк с ошибками. Количество ошибочных строк для вывода ограничено двадцатью, во избежание избыточного объёма вывода на консоль. По итогу чтения файла выводится сводка с количеством считанных записей и строк с ошибками.
- В качестве разделителей полей данных в файле данных кроме “;” могут опционально использоваться запятая “,” , предполагаемая по умолчанию форматом .csv (Comma Separated Values – значения, разделённые запятыми), пробел “ ” или знак табуляции “\t” в любых сочетаниях. При добавлении записей в файл или при перезаписи файла применяются разделители “;”.

- Отображение статистики за год и месяц: последний год в файле данных, явно указанные периоды в виде параметров соответствующих аргументов или текущие.
- При выполнении всех релевантных операций выполняется валидация данных: даты-времени – по формату и значениям с учётом количества дней по месяцам, в т.ч. для високосного года; температуры – по вхождению в диапазон между допустимыми минимальным и максимальным значениями.
- Печать диапазона файла данных.
- Ручное добавление записи в конец файла данных с указанными датой и временем или с текущими значениями даты-времени. В последнем случае в качестве параметра аргумента создания записи указывается только температура. В случае, если файл данных с указанным именем не существует, создаётся новый файл.
- Удаление записи по порядковому номеру с перезаписью файла и корректировкой ошибок – удалением некорректных строк.
- Сортировка записей в файле данных по дате-времени с перезаписью файла и корректировкой ошибок – удалением некорректных строк.

Программа является консольным приложением (CLI) и рекомендуется запускать её из командной строки. Язык интерфейса командной строки – английский.

При запуске без указания ключей, с неподдерживаемыми ключами или ключом «-h» программа выводит справку о поддерживаемых ключах и их назначении.

Программа допускает применение следующих ключей:

- «-h» - получение информации о возможных ключах запуска с кратким описанием их назначения. Также данная информация отображается в случае отсутствия или неверного ввода аргументов и их параметров;
- «-f <filename>» - указание файла для обработки, где <filename> – имя файла. Выводится информация по количеству записей и наличию ошибок в файле, а также сводная статистика по последнему году, данные по которому имеются в файле.
- Ключи «-m [MM]» и «-y [YYYY]» с опциональными параметрами MM и YYYY служат для управления выдаваемой статистикой и указанием периодов выборки данных:
 - «-y [YYYY]» - определяет год для вывода статистики: если ключ отсутствует, принимается последний год, информация по которому присутствует в файле; если ключ присутствует, но параметр YYYY не указан, то принимается текущий год.

- «-m [MM]» - активирует вывод месячной статистики: если параметр MM отсутствует, то статистика выводится за все месяцы года.
- «-p S L» - печать фрагмента данных из файла, длиной L начиная с записи #S.
- «-a [DD MM YYYY HH mm] TT» - добавление записи в конец файла; если указано только значение температуры, запись добавляется с текущими датой/временем. Файл не перезаписывается.
- «-d NN» - удаление записи #NN. Файл перезаписывается полностью с удалением ошибочных строк и форматированием записей в виде YYYY;MM;DD;HH:mm;TT.
- «-s» - сортировка записей в файле по возрастанию даты/времени. Файл перезаписывается полностью с удалением ошибочных строк и форматированием записей в виде YYYY;MM;DD;HH:mm;TT.

Ключи могут применяться совместно в любых сочетаниях. Очередность выполнения операций в случае вызова с несколькими ключами:

1. Чтение файла
2. Добавление записи
3. Удаление записи
4. Сортировка файла
5. Вывод месячной статистики
6. Вывод годовой статистики

Примеры написания командной строки для запуска:

```
run_temp
```

```
run_temp -h
```

```
run_temp -f small_file.csv
```

```
run_temp -f big_file.csv -m 3
```

```
run_temp -f big_file.csv -m 10 -y 2021 -a -10
```

```
run_temp -f big_file.csv -a 2023 10 15 10 0 5 -d 10 -s -m 10 -y
```

Демонстрация работы программы

```
D:\!my\1_MPTI_Education\mipt_hw_C\HW_TEMP_PRJ>run_temp.exe -h
CLI temperature log processing usage:
  -f <filename>: temperature database CSV-file; entry format: <YYYY>;<MM>;<DD>;<HH>;<MM>;<temperature>
                  If the file doesn't exist it will be created if data entry added with -a

Statistics:
  -y [YYYY]: print statistics for year YYYY. Current year is assumed if a value not specified;
              last year in the file is assumed if no argument specified
  -m [MM]: print statistics for month MM. Every month statistics assumed if MM is not specified

Database management:
  -p [start] [length]: print database fragment [length] entries long starting from [start]
  -a <YYYY> <MM> <DD> <HH> <mm> <temperature>: add record to the database file end.
  -a <temperature>: add record to the database with current date/time to the database file end.
  -d <index>: delete record from database by index (#), rewrites the file, deletes inconsistent data
  -s: sort database by date/time, ascending. Rewrites the file, deletes inconsistent data.
```

Рис. 1. Запуск программы с опцией «-h»

```
D:\!my\1_MPTI_Education\mipt_hw_C\HW_TEMP_PRJ>run_temp.exe -f temperature_big.csv -m 10
Reading data from "temperature_big.csv"...
invalid line #83522: 2021; 2;29; 0; 0;29
invalid line #83523: 2021; 2;29; 0; 1;29
invalid line #83524: 2021; 2;29; 0; 2;29
invalid line #83525: 2021; 2;29; 0; 3;29
invalid line #83526: 2021; 2;29; 0; 4;29
invalid line #83527: 2021; 2;29; 0; 5;29
invalid line #83528: 2021; 2;29; 0; 6;29
invalid line #83529: 2021; 2;29; 0; 7;29
invalid line #83530: 2021; 2;29; 0; 8;29
invalid line #83531: 2021; 2;29; 0; 9;29
invalid line #83532: 2021; 2;29; 0;10;29
invalid line #83533: 2021; 2;29; 0;11;29
invalid line #83534: 2021; 2;29; 0;12;29
invalid line #83535: 2021; 2;29; 0;13;29
invalid line #83536: 2021; 2;29; 0;14;29
invalid line #83537: 2021; 2;29; 0;15;29
invalid line #83538: 2021; 2;29; 0;16;29
invalid line #83539: 2021; 2;29; 0;17;29
invalid line #83540: 2021; 2;29; 0;18;29
invalid line #83541: 2021; 2;29; 0;19;29
... and 2860 more.
Done:
      515520 entries read, 2880 invalid lines ignored
Statistics:
*****
October, 2021:
Average temperature: 15
Lowest temperature: 1
Highest temperature: 30
*****
Year 2021:
Average temperature: 14
Lowest temperature: 1
Highest temperature: 30
```

Рис. 2. Запуск программы с указанием файла и опцией выбора месяца. Файл данных содержит 2880 ошибочных записей (несуществующие даты 29 и 30 февраля 2021 г.)


```

D:\!my\1_MPTI_Education\mipt_hw_C\HW_TEMP_PRJ>run_temp.exe -f temperature_big.csv
Reading data from "temperature_big.csv"...
invalid line #83522: 2021; 2;29; 0; 0;29
invalid line #83523: 2021; 2;29; 0; 1;29
invalid line #83524: 2021; 2;29; 0; 2;29
invalid line #83525: 2021; 2;29; 0; 3;29
invalid line #83526: 2021; 2;29; 0; 4;29
invalid line #83527: 2021; 2;29; 0; 5;29
invalid line #83528: 2021; 2;29; 0; 6;29
invalid line #83529: 2021; 2;29; 0; 7;29
invalid line #83530: 2021; 2;29; 0; 8;29
invalid line #83531: 2021; 2;29; 0; 9;29
invalid line #83532: 2021; 2;29; 0;10;29
invalid line #83533: 2021; 2;29; 0;11;29
invalid line #83534: 2021; 2;29; 0;12;29
invalid line #83535: 2021; 2;29; 0;13;29
invalid line #83536: 2021; 2;29; 0;14;29
invalid line #83537: 2021; 2;29; 0;15;29
invalid line #83538: 2021; 2;29; 0;16;29
invalid line #83539: 2021; 2;29; 0;17;29
invalid line #83540: 2021; 2;29; 0;18;29
invalid line #83541: 2021; 2;29; 0;19;29
... and 2860 more.
Done:
      515520 entries read, 2880 invalid lines ignored
Statistics:
*****
Year 2021:
Average temperature: 14
Lowest temperature: 1
Highest temperature: 30

```

Рис. 3. Запуск программы с указанием имени файла.

```

D:\!my\1_MPTI_Education\mipt_hw_C\HW_TEMP_PRJ>run_temp.exe -f temperature_small.csv -p -a 2021 2 18 1 1 -28 -d 5 -s
Reading data from "temperature_small.csv"...
invalid line #5: 2021;01;16;01;05;-xx
Done:
      15 entries read, 1 invalid lines ignored
Entry #16 added: 2021;2;18;1;1;-28
Entry #5 deleted
Database arranged by data/time ascending
50 entries beginning from #1:
*****
| # | Date, time | Temp. |
*****
|001| 2021-01-16 01:01 | -47 |
|002| 2021-01-16 01:03 | -44 |
|003| 2021-01-16 01:04 | -43 |
|004| 2021-02-16 01:01 | -25 |
|005| 2021-02-18 01:01 | -28 |
|006| 2021-03-16 01:01 | -10 |
|007| 2021-04-16 01:01 | 00 |
|008| 2021-05-16 01:01 | 10 |
|009| 2021-06-16 01:01 | 25 |
|010| 2021-07-16 01:01 | 30 |
|011| 2021-08-16 01:01 | 20 |
|012| 2021-09-16 01:01 | 18 |
|013| 2021-10-16 01:01 | 02 |
|014| 2021-11-16 01:01 | -5 |
|015| 2021-12-16 01:01 | -20 |
*****
Statistics:
*****
February, 2021:
Average temperature: -26
Lowest temperature: -28
Highest temperature: -25
*****
Year 2021:
Average temperature: 0
Lowest temperature: -47
Highest temperature: 30

```

Рис. 4. Запуск программы с указанием имени файла, печатью содержимого файла, добавлением записи (выделено), удалением записи с исходным номером #5 и сортировкой записей в файле

Сборка программы утилитой make

```
D:\!my\1_MPTI_Education\mipt_hw_C\HW_TEMP_PRJ>make
gcc -Wall -Wextra -std=c99 -c main.c -o main.o
gcc -Wall -Wextra -std=c99 -c temp_api.c -o temp_api.o
gcc -o run_temp.exe main.o temp_api.o
```

Рис. 5. Вывод утилиты make при сборке программы

Потенциал развития программы

В качестве потенциальных шагов по развитию программы можно предположить следующие функциональные дополнения:

- Слияние данных из нескольких файлов, например, от нескольких датчиков
- Сравнение данных из нескольких файлов
- Анализ адекватности данных для сбора статистики: полнота, равномерность распределения данных по периодам выборки, наличие выбросов и т.д.