Московский авиационный институт   
(государственный технический университет)   
  
Факультет прикладной математики   
  
Кафедра вычислительной математики и программирования

Курсовой проект по курсам

“Языки и методы программирования” и “Алгоритмы и структуры данных”

8 факультет, 2 курс, весенний семестр 2020/21 учебного года

Студент: Стрыгин Д.Д.

Группа: М8О-106Б-19, №20

Преподаватель: Дубинин А.В.

Содержание:

* Введение
* Задание
* Работа с файлами в Си
* Описание программы
* Заключение
* Список источников

Введение

Разработать последовательную структуру данных для представления простейшей базы данных на файлах в СП Си в соответствии с заданным вариантом. Составить программу генерации внешнего нетекстового файла заданной структуры, содержащего представительный набор записей (15-20). Распечатать содержимое сгенерированного файла в виде таблицы и выполнить над ним заданное действие.

Структуры данных и константы, совместно используемые программами, следует вынести в отдельный заголовочный файл.

В процессе отладки и тестирования рекомендуется использовать команды обработки текстовых файлов OC UNIX и переадресацию ввода-вывода. Сгенерированные и отформатированные тестовые данные необходимо заранее поместить в текстовые файлы и распечатывать при протоколировании. Рекомендуется подобрать реальные или правдоподобные тестовые данные. Число наборов тестовых данных должно быть не менее трёх. Имя файла с бинарными данными является обязательным параметром второй программы.

Задание

Посчитать средний балл элементов и вывести те, у которых средний балл выше.

Работа с файлами на Си

Файл – именованная область внешней памяти, выделенная для хранения массива данных. Данные, содержащиеся в файлах, имеют самый разнообразный характер: программы на алгоритмическом или машинном языке; исходные данные для работы программ или результаты выполнения программ; произвольные тексты; графические изображения и т. п.

Файлы в Си используются для того, чтобы сохранять результат работы программы Си и использовать его при новом запуске программы. Например, можно сохранять результаты вычислений, статистику игр. Чтобы работать с файлами в Си необходимо подключить библиотеку stdio.h

Чтобы работать с файлом в Си необходимо задать указатель на файл по образцу

FILE \*имя указателя на файл;

Например

FILE \*fin;

задаёт указатель fin на файл.

В языке Си любой файл рассматривается как поток байтов. Концом любого файла является спец. символ. При открытии файла, ему ставится определенный поток и возвращается указатель на структуру FILE (так же возвращаемое значение именуют - дескриптор файла). Т.е. доступ к элементам файла будет происходить с помощью этого указателя.

FILE \*fopen(name, type);

name – имя открываемого файла (включая путь), type — указатель на строку символов, определяющих способ доступа к файлу:

* "r" — открыть файл для чтения (файл должен существовать);
* "w" — открыть пустой файл для записи; если файл существует, то его содержимое теряется;
* "a" — открыть файл для записи в конец (для добавления); файл создаётся, если он не существует;
* "r+" — открыть файл для чтения и записи (файл должен существовать);
* "w+" — открыть пустой файл для чтения и записи; если файл существует, то его содержимое теряется;
* "a+" — открыть файл для чтения и дополнения, если файл не существует, то он создаётся.

Возвращаемое значение — указатель на открытый поток. Если обнаружена ошибка, то возвращается значение NULL.

Например, следующая команда

file = fopen("C:\\Users\\user\\Desktop\\data.txt", "r");

откроет файл data.txt, находящийся на рабочем столе по пути C:UsersuserDesktop.

В Си путь указывается с помощью двух слешей. После работы с файлом в си, необходимо его закрыть с помощью команды fclose(имя указателя на файл).

**Считывание из файла**

Для считывания слова из файла в Си используется команда fscanf(). Эта команда аналогична команде ввода информации с клавиватуры в Си scanf() только первый параметр это указатель на файл.

fscanf(указатель на файл,"%формат1 %формат2…",&перменная1,&переменная2…);

Например команда

fscanf(file,"%d%d%d",&a,&b,&c);

считает из файла, который привязан к указателю на файл fin строку из трёх целочисленных переменных.

**Запись в файл**

Для записи данных в файл в Си, необходимо открыть файл в режиме записи. Имя указателя на файл= fopen("путь к файлу", "w") Для записи в строку текстового файла используется команда fprnitf(), которая аналогична команде printf() вывод на экран в Cи только первый параметр это указатель на файл.

fprintf (имя указателя на файл,”%формат ввода”, переменные);

Например, запись в файл out.txt значение переменной a

a=10;

file2 = fopen("C:\\Users\\user\\Desktop\\out.txt", "w");

fprintf (file2,”%d”, a);

**Бинарные файлы**

Текстовые файлы хранят данные в виде текста. Это значит, что если, например, мы записываем целое число 12345678 в файл, то записывается 8 символов, а это 8 байт данных несмотря на то, что число помещается в целый тип. Кроме того, вывод и ввод данных является форматированным, то есть каждый раз, когда мы считываем число из файла или записываем в файл происходит трансформация числа в строку или обратно. Это затратные операции, которых можно избежать. Текстовые файлы позволяют хранить информацию в виде, понятном для человека. Можно, однако, хранить данные непосредственно в бинарном виде. Для этих целей используются бинарные файлы.

Запись в файл осуществляется с помощью функции fwrite.

size\_t fwrite ( const void \* ptr, size\_t size, size\_t count, FILE \* stream );

Функция возвращает число удачно записанных элементов. В качестве аргументов принимает указатель на массив, размер одного элемента, число элементов и указатель на файловый поток. Вместо массива, конечно, может быть передан любой объект. Запись в бинарный файл объекта похожа на его отображение: берутся данные из оперативной памяти и пишутся как есть. Для считывания используется функция fread.

size\_t fread ( void \* ptr, size\_t size, size\_t count, FILE \* stream );

Функция возвращает число удачно прочитанных элементов, которые помещаются по адресу ptr. Всего считывается count элементов по size байт.

Для записи в бинарный файл надо открыть его в режиме “wb”, для чтения соответственно “rb”.

Описание программы

Первая программа принимает на вход файл с текстом и файл для записи (если файла для записи не существует, то он создаётся). Программа считывает данные из первого файла как текст и записывает в двоичном виде во второй файл.

Сложность O(n).

Вторая программа работает с бинарным файлом и проходит его дважды: за первый проход вычисляется средний балл, за второй – на экран выводятся все подходящие варианты (те, у которых балл выше среднего).

Сложность O(n), но требуется 2 прохода по файлу.

Заключение

В курсовом проекте были рассмотрены отличия бинарных файлов от текстовых а так же работа с файлами на Си. В ходе подготовки проекта были изучены функции чтения и записи текстовых и бинарных файлов (fread, fprintf, fwrite, fscanf) а так же другие вспомогательные функции (например, rewind) . В процессе выполнения проекта были написаны 2 программы:

1. Программа считывает содержимое файла с таблицей и записывает всё в бинарный файл
2. Программа работает с бинарным файлом, за первый проход вычисляется средний балл, за второй на экран выводятся подходящие варианты (те, у которых балл выше среднего)

Список источников:

* <https://prog-cpp.ru/c-files/> работа с файлами в СИ
* <https://learnc.info/c/binary_files.html> бинарные файлы