

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«Московский Авиационный Институт» (Национальный
Исследовательский Университет)

Институт №8: «Информационные технологии и прикладная математика» Кафедра:
806 «Вычислительная математика и программирование»

КУРСОВОЙ ПРОЕКТ

По курсу «Практикум программирования»

II семестр

Тема:

«Обработка последовательной файловой структуры»

Группа:	М8О-106Б-22
Студент:	Голубев Т.Д.
Преподаватель:	Дубинин А.В.
Оценка:	
Дата:	

Москва, 2022

Оглавление

Оглавление

Оглавление.....	2
Постановка задачи.....	3
Теоретическая часть.....	4
Описание алгоритма решения.....	5
Вывод.....	6

Постановка задачи

Разработать последовательную структуру данных для представления простейшей базы данных на файлах в СП Си. База данных содержит сведения о составе комплектующих личных ПЭВМ в студенческой группе: фамилия владельца, производитель процессора, количество ядер, тип видеокарты, размер видеопамати, название операционной системы.

Составить программу, которая печатает список однофамильцев, имеющих однотипные компьютеры (производитель процессора, количество ядер, тип видеокарты, размер видеопамати и название операционной системы совпадают).

Теоретическая часть

Для решения задачи необходимо прочесть файл. Работа с файлами в Си осуществляется с помощью файловых дескрипторов.

Файловый дескриптор — это неотрицательное число, которое является идентификатором потока ввода-вывода. Дескриптор может быть связан с файлом, каталогом, сокетом.

Обычно файловые дескрипторы выделяются последовательно. Есть пул свободных номеров. Когда создаётся новый файл или открывается существующий, ему присваивается номер.

Дескриптор для каждого процесса является уникальным. Но есть три жестко закрепленных индекса — это первые три номера (0, 1, 2).

- 0 — стандартный ввод (stdin), место, из которого программа получает интерактивный ввод.
- 1 — стандартный вывод (stdout), на который направлена большая часть вывода программы.
- 2 — стандартный поток ошибок (stderr), в который направляются сообщения об ошибках.

Когда вы завершаете работу с файлом, присвоенный ему дескриптор освобождается и возвращается в пул свободных номеров. Он снова доступен для выделения под новый файл.

Описание алгоритма решения

С помощью первой программы читаем базу данных с помощью функции `fscanf`, которая представлена текстовым файлом. Записываем все поля БД в структуру и помещаем эту структуру в бинарный файл с помощью функции `fread`. В начало бинарного файла записываем количество строк в БД. Асимптотическая сложность — $O(n)$.

С помощью второй пробегаем по бинарному файлу двумя вложенными циклами (так как файл может быть не отсортирован) и ищем однофамильцев с однотипными компьютерами, сравнивая все поля двух структур. Сложность — $O(n^2)$ из-за вложенных циклов.

Навигация по бинарному файлу осуществляется с помощью функции `fseek`. Она позволяет передвинуть указатель на заданное количество байт от начала (`SEEK_SET`), конца (`SEEK_END`) или текущего положения (`SEEK_CUR`).

Вывод

В ходе работы я научился работать с файлами в Си, узнал про файловые дескрипторы и бинарные файлы; узнал про преимущества бинарного файла перед обычным текстовым (в первом возможна удобная навигация с помощью `fseek`).