

## Функции и файлы

### Вариант 1

Для хранения данных о планшетных сканерах описать структуру вида:

```
struct scan_info{
    char model[25];        // наименование модели
    int price;             // цена
    double x_size;         // горизонтальный размер области сканирования
    double y_size;         // вертикальный размер области сканирования
    int optr;              // оптическое разрешение
    int grey;              // число градаций серого
};
```

Написать функцию, которая записывает в бинарный файл данные о сканере из приведенной структуры. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая извлекает из этого файла данные о сканере в структуру типа `scan_info`. Обязательный параметр — номер требуемой записи. Функция должна возвращать нулевое значение, если чтение прошло успешно, и `-1` в противном случае.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) — 6–8 записей и выводящей на дисплей данные о запрошенной записи.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

### Вариант 2

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая записывает в бинарный файл данные о сканере из приведенной структуры. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая сортирует записи в описанном выше бинарном файле по одной из следующих характеристик: цена либо число градаций серого. Обязательный параметр — признак, задающий критерий сортировки.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) из не менее восьми записей и осуществляющий его сортировку.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

### Вариант 3

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая записывает в бинарный файл данные о сканере из приведенной структуры. Структура файла: в первых четырех байтах размещается значение типа `long`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая сортирует записи в описанном выше бинарном файле по наименованию модели сканера.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) из не менее восьми записей и осуществляющий его сортировку.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

### Вариант 4

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая динамически выделяет память под массив структур (не меньше шести элементов), заполняет его данными в режиме диалога и записывает массив в бинарный файл. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая извлекает данные о сканере из описанного выше бинарного файла в структуру типа `scan_info`. Обязательный параметр — номер требуемой записи. Функция должна возвращать нулевое значение, если чтение прошло успешно, и `-1` в противном случае.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) из не менее восьми записей и осуществляющий вывод на дисплей данных о требуемой записи.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

## Вариант 5

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая записывает данные о сканере из приведенной структуры в требуемую позицию в бинарном файле. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах. Запись может осуществляться в любую позицию, причем если между вводимой записью и последней (или началом файла) имеются пропуски, они заполняются нулями.

Написать функцию, которая «уплотняет» описанный выше бинарный файл путем удаления из него записей, содержащих все нули.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) из не менее шести записей и осуществляющий его уплотнение.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

## Вариант 6

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая динамически выделяет память под массив структур (не меньше шести элементов), заполняет его данными в режиме диалога и записывает массив в бинарный файл. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая запрашивает данные о сканере в режиме диалога и замещает записи в бинарном файле по заданному номеру. Обязательный параметр — номер замещаемой записи. Функция должна возвращать нулевое значение, если запись прошла успешно, и `-1` в противном случае.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся с клавиатуры) из не менее восьми записей и осуществляющий вставку новых данных о сканере.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

## Вариант 7

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая записывает в бинарный файл данные о сканере из приведенной структуры. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая вводит данные о сканере с клавиатуры в структуру типа `scan_info`, и если данные об этом сканере отсутствуют в файле, помещает со-

держимое структуры в конец файла; в противном случае выдает соответствующее сообщение.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся из текстового файла) — 6–8 записей и дополняющей файл записями о 2–3 сканерах, вводимых с клавиатуры.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

## Вариант 8

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая записывает в бинарный файл данные о сканере из приведенной структуры. Структура файла: в первых двух байтах размещается значение типа `int`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Написать функцию, которая вводит данные о сканере с клавиатуры в структуру типа `scan_info` и помещает ее содержимое на место первой записи в файле. Файл должен существовать. При этом, запись ранее занимавшая первую позицию, помещается на вторую, вторая запись на третью, и т. д.

Привести пример программы, создающей файл с данными о сканерах (данные вводятся из текстового файла) — 6–8 записей и дополняющей этот файл 1–2 новыми записями, вводимыми с клавиатуры.

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

## Вариант 9

Для хранения данных о планшетных сканерах описать структуру вида, описанного в варианте 1.

Написать функцию, которая запрашивает количество сканеров, информация о которых будет вводиться, динамически выделяет память под массив структур соответствующего размера и заполняет его данными в режиме диалога (с клавиатуры). При этом имя сканера может содержать пробелы.

Написать функцию, которая записывает данный массив в создаваемый бинарный файл. Если цена сканера меньше 200, то данные об этом сканере в файл не записываются. Информация об остальных сканерах помещается в бинарный файл, причем сначала пишутся данные о всех сканерах, имя которых начинается с заглавной буквы, а затем — с прописной.

Структура файла: в первых четырех байтах размещается значение типа `long`, определяющее количество сделанных в файл записей; далее без пропусков размещаются записи о сканерах.

Привести пример программы, создающей файл с данными о сканерах и осуществляющий вывод на дисплей данных о требуемой записи (либо всех, либо по номеру).

Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.

## Вариант 10

Для хранения данных о ноутбуках описать структуру вида (при необходимости дополнив ее):

```
struct NOTEBOOK{
    char model[21]; // наименование
    struct size{    // габаритные размеры
        float x;
        float y;
        float z;
    };
    float w;        // вес
    int price;      // цена
}
```

Написать функцию, которая читает данные о ноутбуках из файла note.txt (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта (целое) — число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется чтение данных только для тех ноутбуков, частота процессора которых больше 120 МГц, и запись в бинарный файл по убыванию цены.

## Вариант 11

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 10.

Написать функцию, которая читает данные о ноутбуках из файла note.txt (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта (целое) — число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется чтение данных только для тех ноутбуков, объем HDD которых меньше 1 Гбайт, и запись считанных данных в бинарный файл в алфавитном порядке по наименованию.

## Вариант 12

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 10.

Написать функцию, которая читает данные о ноутбуках из файла note.txt (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта (целое) — число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, целое количество которых в одном кубическом метре не превышает 285 штук.

### Вариант 13

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 10.

Написать функцию, которая читает данные о ноутбуках из файла `note.txt` (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта (целое) — число записей в файле; далее записи в формате структуры `NOTEBOOK`.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, максимальный объем ОЗУ которых не менее 40 Мбайт, отсортированных по объему.

### Вариант 14

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 10.

Написать функцию, которая читает данные о ноутбуках из файла `note.txt` (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта — целое число записей в файле; далее записи в формате структуры `NOTEBOOK`.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, диагональ дисплея которых больше одиннадцати дюймов.

### Вариант 15

Для хранения данных о ноутбуках описать структуру вида (при необходимости дополнив ее):

```
struct NOTEBOOK{
    struct disp_res{    // разрешающая способность дисплея
        int x;          // по горизонтали
        int y;          // по вертикали
    };
    int f;              // частота регенерации
    float d;            // размер диагонали дисплея
    int price;          // цена
    char model[21];     // наименование
}
```

Написать функцию, которая читает данные о ноутбуках из файла `note.txt` (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла:

первые два байта — целое число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, вес которых менее 7 кг, отсортированных в порядке возрастания цены.

### Вариант 16

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 15.

Написать функцию, которая читает данные о ноутбуках из файла note.txt (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта — целое число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, объем видеопамяти которых 2 Мбайт, отсортированных в порядке уменьшения тактовой частоты процессора.

### Вариант 17

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 15.

Написать функцию, которая читает данные о ноутбуках из файла note.txt (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта — целое число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, объем HDD которых больше 1 Гбайт, отсортированных в порядке возрастания размера диагонали дисплея.

### Вариант 18

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 15.

Написать функцию, которая читает данные о ноутбуках из файла note.txt (см. с. 158) в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта — целое число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, тактовая частота процессора которых больше 120МГц, отсортированных в порядке уменьшения веса.

## Вариант 19

Для хранения данных о ноутбуках описать структуру вида (при необходимости дополнив ее):

```
struct NOTEBOOK{
    struct disp_res{ // разрешающая способность дисплея
        int x;        // по горизонтали
        int y;        // по вертикали
    };
    int f;            // частота регенерации
    float d;          // размер диагонали дисплея
    float hdd;        // объем диска
    char model[21];   // наименование
}
```

Написать функцию, которая читает данные о ноутбуках из файла note.txt в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта — целое число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, тактовая частота процессора которых больше 120МГц, отсортированные в порядке возрастания цены.

## Вариант 20

Для хранения данных о ноутбуках описать структуру вида, описанного в варианте 19.

Написать функцию, которая читает данные о ноутбуках из файла note.txt в структуру приведенного вида. Написать функцию, которая записывает содержимое структуры в конец бинарного файла. Структура бинарного файла: первые два байта — целое число записей в файле; далее записи в формате структуры NOTEBOOK.

Написать программу, в которой на основе разработанных функций осуществляется запись в двоичный файл данных только о тех ноутбуках, цена которых больше \$3500, отсортированные в порядке возрастания тактовой частоты процессора.

Пример файла note.txt:

Acer Note Light	2699	5.6	02.0x11.8x08.3	100	40	10.4	1	1024x0768	60	0.774
ASW ND5123T	3489	7.2	02.3x11.8x10.1	133	32	12.1	2	1024x0768	70	1.300
ARMNote TS80CD	3699	7.2	02.0x11.5x08.8	133	64	11.3	1	1024x0768	75	1.300
AST Ascentia P50	4499	7.5	02.3x11.3x09.0	133	40	11.3	1	0800x0600	70	0.774
BSI NP8657D	2605	8.0	02.3x11.8x09.3	133	40	11.3	1	1024x0768	60	0.810
BSI NP5265A	3765	8.2	02.5x12.0x09.0	150	32	12.1	2	1024x0768	70	1.300
Dell Xpi P100SD	3459	6.0	02.3x11.0x08.8	100	40	10.3	1	1024x0768	60	0.773
Digital HiNote	4799	4.0	01.3x11.0x08.8	120	40	10.4	1	0800x0600	56	1.000
Gateway Solo S5	4499	5.6	02.0x11.9x08.8	133	40	11.3	2	1024x0768	60	0.686
Hertz Z-Optima NB	3995	8.0	02.3x11.9x09.0	150	40	11.2	2	1024x0768	75	1.000
HP OmniBook 5500	6120	7.1	02.0x11.5x09.0	133	64	11.4	1	1024x0768	75	1.300
IBM ThinkPad 560	3749	4.1	01.3x11.8x08.8	120	40	12.1	2	1024x0768	85	0.774
NEC Versa 4080H	4780	6.6	02.3x11.8x09.5	120	48	10.4	1	0800x0600	70	0.776
Polywell Poly 500	3300	7.9	02.3x11.9x09.0	120	40	10.4	1	1024x0768	72	1.000



```
Samsung SENS 810 3667 8.7 02.3x11.5x09.5 100 32 11.4 2 1024x0768 75 0.773
Twinhead Slimnote 2965 7.4 02.0x11.5x08.0 075 64 10.4 1 1024x0768 70 0.772
```

В файле note.txt находится текстовая информация о ноутбуках. Каждая строка содержит данные об одной модели. Данные в строке размещаются в следующих полях:

- 1 : 20 — наименование модели;
- 21 : 24 — цена в долларах (целое число);
- 26 : 28 — масса ноутбука в кг (число с десятичной точкой из четырех символов);
- 30 : 43 — габаритные размеры ноутбука в дюймах (ВЫСОТАxДЛИНАxШИРИНА — три числа с десятичной точкой (4 символа, включая точку, разделенные 'x');
- 44 : 47 — частота процессора в МГц (целое число из трех символов);
- 49 : 50 — максимальный объем ОЗУ в мегабайтах (целое число из двух символов);
- 52 : 55 — размер диагонали дисплея в дюймах (число с десятичной точкой из четырех символов, включая точку);
- 57 — размер видеопамати в мегабайтах — целое число из одного символа;
- 59 : 67 — разрешающая способность дисплея в пикселах (два целых числа, разделенные 'x');
- 69 : 70 — частота регенерации дисплея в Гц (целое число из двух символов);
- 72 : 76 — объем HDD в гигабайтах (число с десятичной точкой из пяти символов).

Все неописанные позиции заполнены пробелами.