

1. Дана квадратная матрица A порядка M . Начиная с первого элемента первой строки, вывести ее элементы следующим образом:

все элементы первого столбца;
элементы последней строки, кроме первого (уже выведенного) элемента;
оставшиеся элементы второго столбца;
оставшиеся элементы предпоследней строки;
и т. д.;

последний элемент первой строки.
2. Дана целочисленная матрица размера $M \times N$. Найти номер первой из ее строк, содержащих равное количество положительных и отрицательных элементов (нулевые элементы матрицы не учитываются). Если таких строк нет, то вывести 0.
3. Элемент матрицы называется ее локальным минимумом, если он меньше всех окружающих его элементов. Заменить все локальные минимумы данной матрицы на нули. При решении допускается использовать вспомогательную матрицу.
4. Дана квадратная матрица A порядка M . Найти сумму элементов каждой ее диагонали, параллельной побочной (начиная с одноэлементной диагонали A_{11}).
5. Дана строка символов. Подсчитать число символов в наиболее длинной группе цифр, встречающейся в строке.
6. Дана строка, содержащая, по крайней мере, один символ пробела. Вывести подстроку, расположенную между первым и вторым пробелом исходной строки. Если строка содержит только один пробел, то вывести пустую строку.
7. Определить структуру для хранения угла в градусной мере: {градусы, минуты, секунды}. Написать функцию `Grd_to_Rd(угол в градусной мере)`, переводящую угол из градусов, минут, секунд в радианы и функцию `Rd_to_Grd(угол в радианной мере)`, выполняющую обратное преобразование. Используя эти функции, сложить два угла, заданных в градусах, минутах, секундах, и результат вывести на экран в таком же формате.
8. Массив записей с именем `NOTE`, содержит сведения о знакомых: {Фамилия Имя; Номер телефона; День рождения (*массив из трёх чисел*)}. Написать программу, обеспечивающую ввод с клавиатуры данных в массив `NOTE` и вывод на экран информации о человеке, номер телефона которого введен с клавиатуры. Если такого нет, выдать на дисплей соответствующее сообщение.
9. Дан файл F , записи которого есть целые числа, не равные нулю. Переписать его в текстовый файл, упорядочив значения по возрастанию.

10. 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. ниже) 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.

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

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 (ВЫСОТАхДЛИНАхШИРИНА — 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).

Пример файла note.txt: (см. сл. стр. Можно скопировать в Блокнот и сохранить как note.txt)

N3 17

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 PIOOSD	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 Omni Book 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