

## Задание1

В качестве задания необходимо выполнить общую часть и на выбор вариант 1 либо Вариант 2

### Общая часть

Для следующей задачи реализовать 2 программы: с использованием низкоуровневого ввода вывода и отображения в память.

Написать программу перемножения 2х комплексный матриц.

Предварительно создать файл в котором матрица хранится следующим образом: первые два числа типа unsigned int это размер матрицы (количество строк и количество столбцов) остальные данные это элементы матрицы типа struct compl{ float re; float im;};

В командной строке передаются имена 2 файлов в каждом из которых хранится одна матрица. Программа записывает результат в файл с именем filres.dat и в стандартный поток вывода.

### Вариант1

Написать программу, которая реализует команду ls

ls [-a -l -R] [файл]

все задается в командной строке например: ./a.out ls -l file1

Опции в командной строке могут появляться в произвольном порядке и количестве.

[ ] -означают, что этих параметров может и не быть например:

ls

ls -a

ls -l myfile

ls -a -l dir

...

Если в качестве файла задано имя каталога, то выводится информация обо всех содержащихся в нём файлах, если имя файла, то выводится информация только об этом файле. По умолчанию выводится информация о файлах текущего каталога.

1. -a – вывести имена всех файлов (т.е. и тех, имена которых начинаются с точки);
2. -l - вывести подробную информацию о файлах. Имя файла, тип файла, права доступа, число ссылок на файл, размер файла в байтах - обязательно, имена владельца файла и владеющей группы, дата и время последнего изменения – по желанию (необязательно);
3. -R – рекурсивно обойти встретившиеся подкаталоги (вывести содержимое подкаталогов всех уровней. Например, если в каталоге есть подкаталог Dir1 и Dir2 то ls -R выводит

.

..

Dir1 Dir2 file1 file2

./Dir1:

file1d1 file2d1

./Dir2:

file1d2 file2d2

## Вариант 2

1. Даны два бинарных файла, содержащие последовательности целых чисел, упорядоченных по возрастанию. Объединить их в один файл, также упорядоченный по возрастанию. Имена исходных файлов и файла, в который должен быть записан результат, передаются в командной строке.
2. В командной строке передается имя файла и символ. Удалить из файла все строки, содержащие (не содержащие) заданный символ.
  - а) Длина строки в файле ограничена.
  - б) Длина строки в файле не ограничена.
3. а) Написать функцию, которая читает со стандартного ввода по слову и создает список слов. Конец чтения фиксируется, когда достигнут перевод строки “\n”(enter) или конец файла(ctrl+D). Возвращает указатель на начало списка.  
б) Сделать то же, но для чтения из заданного файла. Файл содержит последовательность слов, читать слова только из строки файла с заданным номером. Если строки с таким номером нет, вернуть NULL. Имя файла и номер строки передаются в командной строке.  
Написать программу, в которой вызывается эта функция (эти функции) и полученный список слов выводится на печать в обратном порядке.  
(Слово — последовательность символов, ограниченная пробелами, символом перевода строки, началом или концом файла.  
Обратите внимание, что в обратном порядке выводятся именно слова, а не символы строки.  
Обеспечьте корректную выполнение при работе с пустым файлом, с пустой строкой. По окончании работы освободите всю динамически выделенную память.)
4. Программа. В командной строке передается имя файла и число N. Если длина файла не больше 1024 байт, то продублировать в этом файле каждую группу из N байт (остаток байт <N в файл записать, но не дублировать). Если длина файла больше 1024, файл не изменять. Указания к задаче Длину файла можно узнать а) с помощью функции stat(), определив значение поля st\_size в struct stat. При этом не требуется открывать файл. б) с помощью функции lseek(), в этом случае файл должен быть предварительно открыт .
5. Программа. В командной строке передается имя файла. Если у файла есть только одно имя (т.е. одна жесткая ссылка), преобразовать файл, «перевернув» его (1-ый символ становится последним, 2-ой — предпоследним и т.д.) без использования вспомогательных файлов. Запрещается также считывать весь файл в память. Указания к задаче Количество жестких ссылок на файл можно получить с помощью функции stat(), определив значение поля st\_nlink в struct stat.
6. Реализовать команду UNIX  
cp [-ip] file copyoffile // копирование файла.  
все задается в командной строке например: ./a.out cp -i file1 file2  
Копируется содержимое исходного файла file (или указуемого исходным, если последний представляет собой символическую ссылку) в целевой copyoffile.  
Копирование в себя является ошибкой (должно отслеживаться).  
Замечание: Копирование осуществлять блоками, а не по 1 байту.  
Опции:  
-i – запрос подтверждения перед попыткой копирования в существующий файл,  
-r – вызывает копирование не только содержимого, но и атрибутов исходного файла (режима, идентификаторов владельца и владеющей группы, времени

последнего доступа и изменения). Достаточно реализовать копирование режима доступа. Порядок опций при задании команды может быть любым, и любая из них (в том числе и обе) может отсутствовать.

Указания Для изменения режима доступа использовать одну из функций:

```
int chmod (const char* path, mode_t mode);
```

```
int fchmod (int filedes, mode_t mode);
```

Необходимо подключить заголовочные файлы <sys/types.h> и <sys/stat.h>.