

# Программирование на языке С

## Модуль 10.

# ФАЙЛЫ

- Понятие потока – stream
- Структура типа FILE
- Открытие и закрытие файлов
- Операции ввода-вывода
- Указатель чтения-записи в файле
- Последовательный и произвольный доступ



## Поддержка ввода-вывода в Си.

- **Отсутствие операторов ввода-вывода** — все операции ввода-вывода осуществляются посредством стандартных функций
- **Отсутствие поддержки внутренней организации файлов** — файлы трактуются как неструктурированные последовательности символов, что в большинстве случаев позволяет переносить механизмы работы с ними на устройства ввода-вывода информации и процессы в оперативной памяти
- Поддержка **потокowego** (побайтового, библиотека `stdio.h`) и **низкоуровневого** ввода-вывода данных (`io.h`)
- Потокový обмен с файлами подразумевает прозрачную буферизацию операций. Низкоуровневый обмен наиболее близок к работе операционной системы



# Стандартные потоки (начало).

- **Стандартные потоки**

- поток ввода **stdin** (по умолчанию — клавиатура)
- поток вывода **stdout** (по умолчанию — экран)
- поток вывода сообщений об ошибках **stderr** (по умолчанию — экран)

- **Открытие потока**

1. Создать (объявить) указатель на файл (тип указателя **FILE**).

2. Открыть (попытаться открыть) поток

Для этого используется функция: **fopen**("<путь\имя файла>", "<режим>");

Возвращает указатель на файл.

"r" Файл открывается для чтения. Если файл не найден, то функция «терпит неудачу» "w" Открывает пустой файл для записи. Если файл существует, то его содержимое уничтожается (БЕЗ ПРЕДУПРЕЖДЕНИЯ!!!) "a" Открывает файл для дозаписи в конец файла. "r+" Файл открывается для чтения с возможность записи в него. Если файл не найден, то функция «терпит неудачу» "w+" Файл открывается для записи с возможность чтения из него. Если файл существует, то его содержимое уничтожается (БЕЗ ПРЕДУПРЕЖДЕНИЯ!!!) "a+" Открывает файл для дозаписи в конец файла с возможностью чтения из него.

В «режиме» могут использоваться также символы b и t (по умолчанию), что говорит о БИНАРНОМ и ТЕКСТОВОМ (соответственно) режиме работы с файлом. Например

"wb+" "rt"

## Стандартные потоки (окончание).

- 3. Проверить успешность выполнения этапа №2 (УКАЗАТЕЛЬ НА ФАЙЛ ДОЛЖЕН БЫТЬ НЕ NULL).
- 4. Если этап №2 выполнен успешно (т.е. файл открыт), то можно работать с файлом - ЗАПИСЫВАТЬ и ЧИТАТЬ данные:
  - 1. Форматный ввод/вывод: `fprintf` - `fscanf`  
`fputs` - `fgets`  
`fputc` - `fgetc`
  - 2. Неформатный ввод/вывод: `fwrite` - `fread`
- 5. Заккрытие потока  
`fclose(FILE *)`

## Перечень основных функций (начало).

- **Ввод-вывод символов (символьный ввод-вывод)**

`int getchar()` — возвращает введенный символ или признак EOF

`int putchar(int)` — возвращает выведенный символ

`int getc(FILE*)` — осуществляет ввод из потока

`int putc(int, FILE*)` — осуществляет вывод в поток

- **Вывод строк (строковый ввод-вывод)**

`char *gets(char *)` — возвращает введенную строку или NULL

`int puts(char *)` — возвращает последний выведенный символ или EOF



## Перечень основных функций (окончание).

- **Форматный ввод-вывод**

`int scanf(const char *, ...)` — возвращает количество введенных объектов или EOF при достижении конца файла либо -1 как признак ошибки преобразования данных

`int printf(const char *, ...)` — возвращает количество введенных символов или отрицательное значение как признак ошибки

- **Функции с параметром для задания потока**

`fgetc fputc fprintf fscanf fgets fputs`

- **Позиционирование в потоке**

`int fseek(FILE*, long <смещение>, int<точка отсчета>)`

- **Вывод сообщения об ошибке в поток stderr**

`perror(<сообщение>)`

## Практика / ДЗ

- Чтение текстового файла.
- Ввод-вывод числовых данных из файла.
- Дано: текстовый файл с положительными числами.  
Реализовать ввод данных из файла с проверкой, что читаемое число имеет корректный формат.





# Список литературы

- [Кнут08] Кнут Д.Э. Искусство программирования / Пер. с англ. — Т. 3. Сортировка и поиск. — 2-е изд. — М.: Вильямс, 2008. — 824 с.
- [КР92] Керниган Б., Ритчи Д. Язык программирования Си / Пер. с англ. — М.: Финансы и статистика, 1992. — 272 с.
- [КР06] Керниган Б., Ритчи Д. Язык программирования С / Пер. с англ. — М.: Вильямс, 2006. — 304 с.
- [Под04] Подбельский В.В., Фомин С.С. Программирование на языке Си. – 2-е доп. изд. – М., Финансы и статистика, 2004. – 600 с.
- [Уэз82] Уэзерелл Ч. Этюды для программистов / Пер. с англ. — М.: Мир, 1982. — 288 с.

