

# Работа с файлами

## Аргументы командной строки

# Еще раз про память

Как уже говорилось, в компьютере выделяют память **постоянную, оперативную и память процессора.**

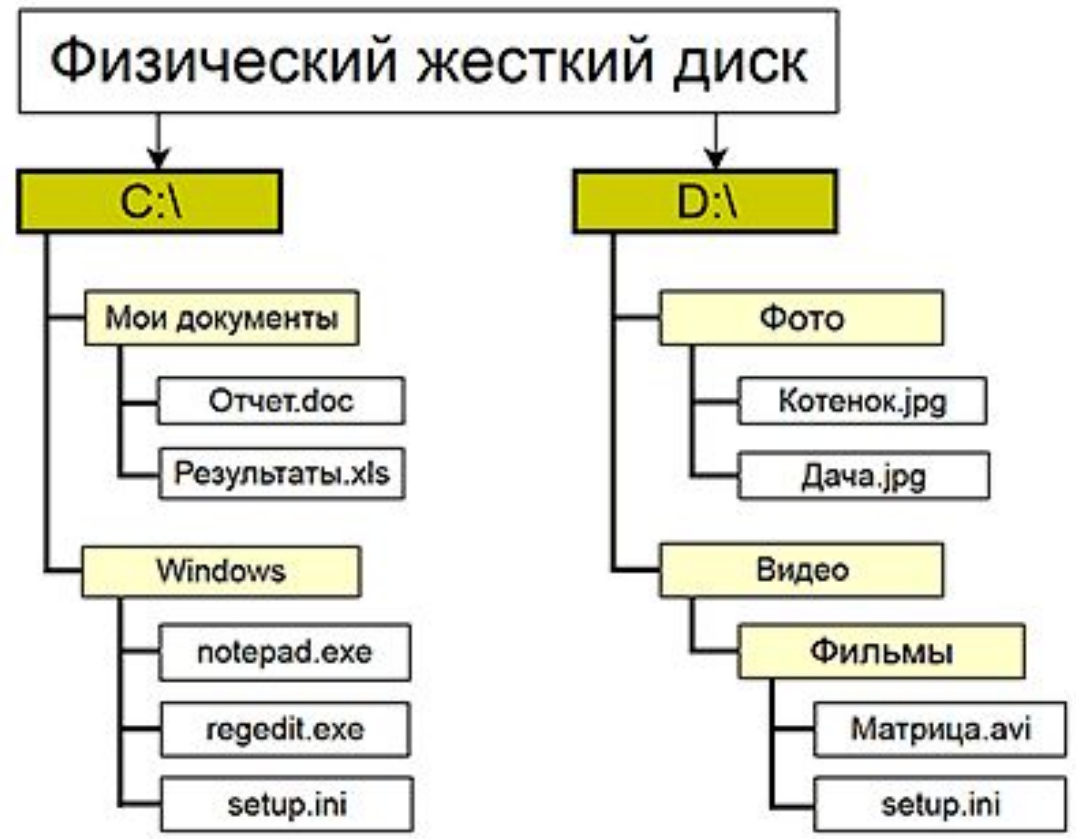
- ▶ Для взаимодействия с **оперативной памятью** используются средства выделения и очистки памяти.
- ▶ **Память процессора** представлена в основном **регистрами и стеком**, с последним связано понятие **локальных переменных**.
- ▶ **Постоянная память** же представлена некоторым **накопительным устройством**, вроде дисков, **flash-памяти** и т.д. Взаимодействие с ней предоставляет операционная система.

# Файловая система

Для удобного использования постоянной памяти, ее представляют в виде некоторой структуры, как правило, древовидной, где каждый узел задает каталог, а лист - файл. Структура, которой задана память в ПЗУ, называется файловой системой.

Некоторые ФС:

- ▶ Fat, Fat32, exFat, NTFS - используются в Windows
- ▶ ext, ext2, ext3, ext4 - Linux
- ▶ HFS, HFS+, APFS - macOS



# Файлы



Работа с файлами.pptx

13.10.2017 23:39

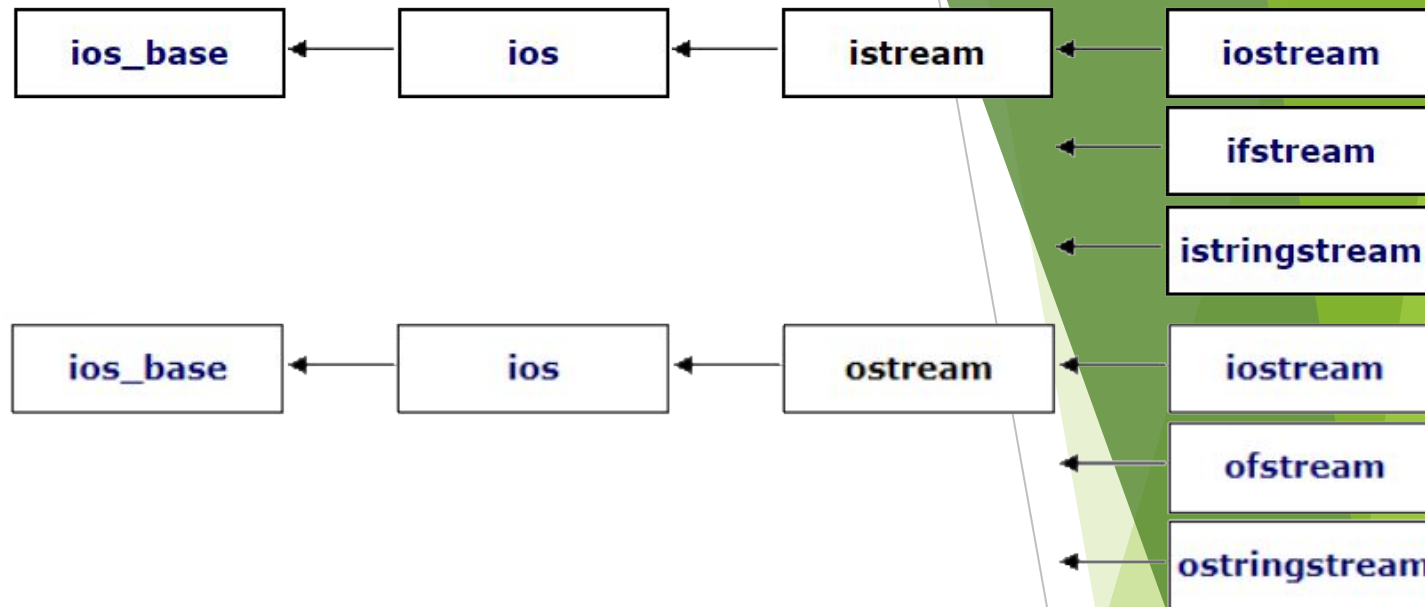
Презентация Мис...

132 КБ

Для любого файла характерны:

- ▶ Имя
- ▶ Расширение - часть имени файла, которая идет после точки (.cpp, .exe)
- ▶ Атрибуты - установки для файла. Например, права доступа, скрытость.
- ▶ Дата изменения
- ▶ Размер

## Потоки данных в C++



Для упрощения работы с потоками данных существует класс `ios_base`. От него путем наследования происходят `istream` и `ostream`. Эти классы представляют собой универсальные классы для потоковой работы с данными. `cin` и `cout` - объекты этих классов для ввода-вывода.

Далее от классов `istream` и `ostream` наследуются классы `ifstream` и `ofstream`. Их принцип работы устроен точно также, но в качестве потока используются определенные файлы.

# Input file stream

Файл с помощью ifstream можно открыть двумя способами:

- ▶ При создании объекта ifstream

```
1  #include <fstream>
2  std::ifstream input("input.txt");
```

- ▶ С помощью функции open

```
1  #include <fstream>
2  std::ifstream input;
3  input.open("input.txt");
```

# Output file stream

Файл с помощью ofstream можно открыть двумя способами:

- ▶ При создании объекта ofstream

```
1  #include <fstream>
2  std::ofstream output("output.txt");
```

- ▶ С помощью функции open

```
1  #include <fstream>
2  std::ofstream output;
3  output.open("output.txt");
```

# File stream

Классы `ifstream` и `ofstream` представлены в качестве одного - `fstream`. По умолчанию он открывает файл в режиме ввода-вывода, но всегда можно указать вторым аргументом режим работы (для `ifstream` и `ofstream` нельзя использовать противоречивые режимы работы).

```
1 #include <fstream>
2 std::fstream file("text.txt", std::ios_base::in | std::ios_base::out);
3 std::fstream bin("file.bin", std::ios_base::in | std::ios_base::out | std::ios_base::binary);
```



# Методы классов

Здесь рассмотрены только некоторые основные методы этих классов.

Все методы представлены здесь:

<http://www.cplusplus.com/reference/fstream/fstream/>

Сигнатура	ifstream	ofstream	Описание
<code>void open(const char* filename, ios_base::openmode mode)</code>	+	+	Открывает файл <b>filename</b> с режимом работы <b>mode</b> .
<code>bool is_open()</code>	+	+	Вернет <b>true</b> , если файл открыт.
<code>void close()</code>	+	+	Закрывает файл.
<code>operator &gt;&gt;</code>	+	-	Осуществляет потоковый ввод из файла.
<code>int get()</code>	+	-	Возвращает символ из файла.
<code>istream&amp; read(char* s, streamsize n)</code>	+	-	Читает массив символов размера <b>n</b> .
<code>operator &lt;&lt;</code>	-	+	Осуществляет потоковый вывод в файл.
<code>ostream&amp; put(char c)</code>	-	+	Записывает символ в файл.
<code>ostream&amp; write(const char* s, streamsize n)</code>	-	+	Записывает массив символов размера <b>n</b> .