

# ПРОГРАММИРОВАНИЕ. Практика

## Тема 2. Основы C++



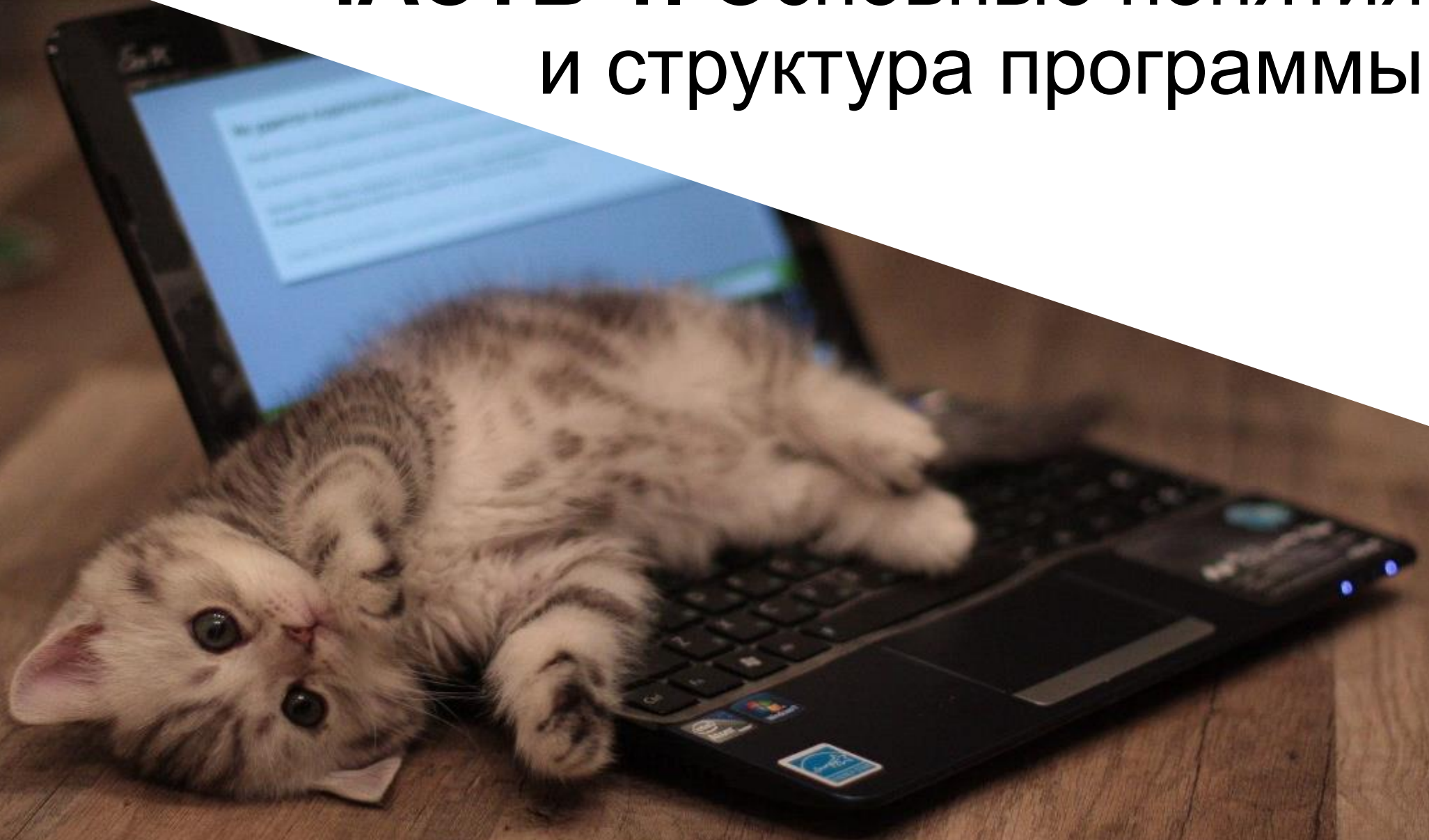
**ФЕДОТЕНКО**

**Мария Александровна**

 [ma.fedotenko@mpgu.su](mailto:ma.fedotenko@mpgu.su)

 [marusyafm](#)

# ЧАСТЬ 1. Основные понятия и структура программы



# Программа Hello world



```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     cout << "Hello, world!";
6 }
```

# Структура программы

- Подключение библиотек и директив препроцессора
- Объявление глобальных переменных
- Функции

```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     cout << "Hello, world!";
6 }
```

# Подключение библиотек и директив

**Директивы препроцессора** – определяют действия по преобразованию программы перед компиляцией, а также включают инструкции, которым компилятор следует во время компиляции

Препроцессор – программа, действующая как фильтр на этапе компиляции. Перед тем, как попасть на вход компилятора, исходная программа проходит через препроцессор.

Можно подключать свои файлы

```
1 ▾ #include <iostream>
2   #include <math.h>
3   #include "file2.cpp"
4   using namespace std;
```

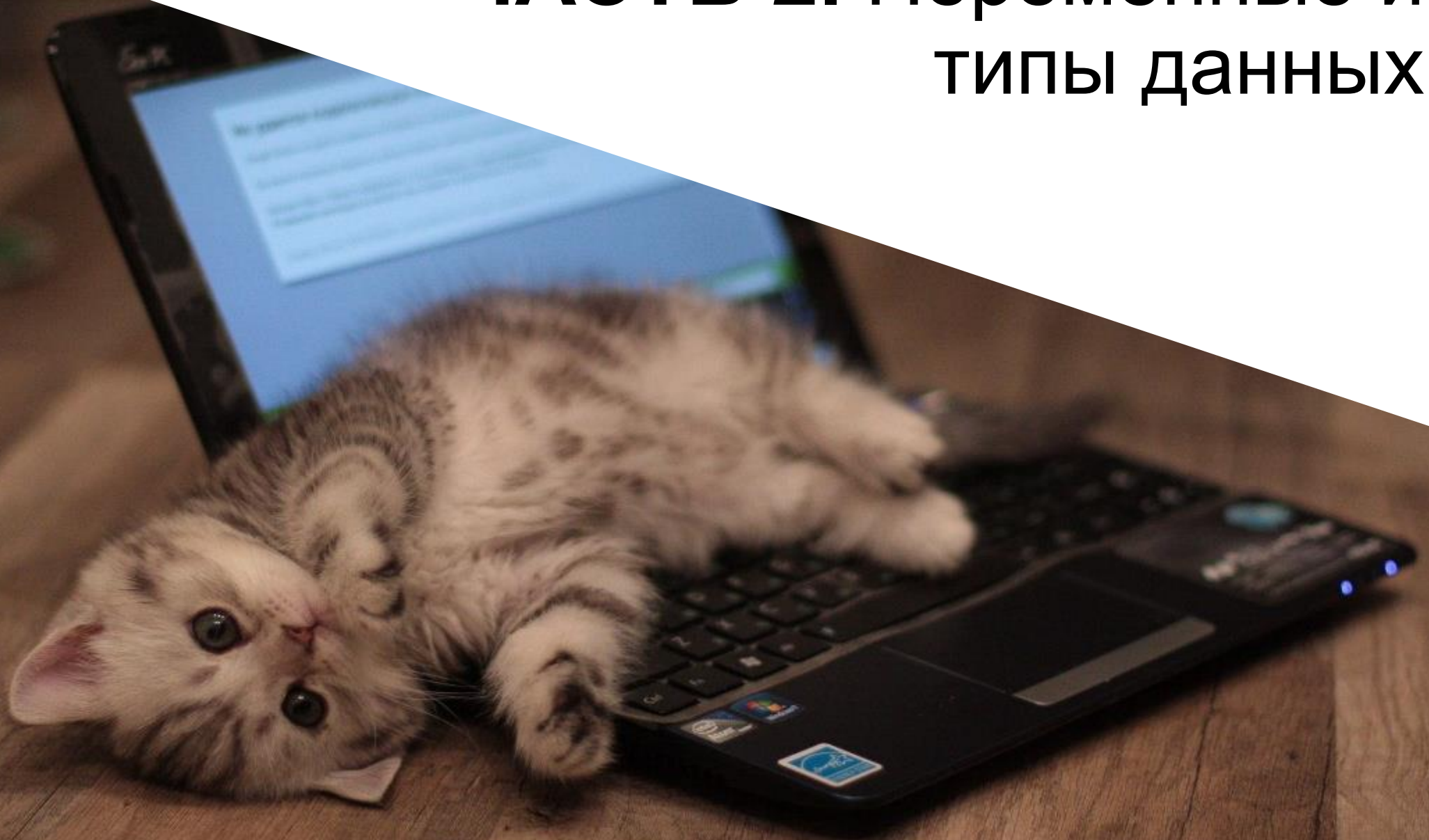
# Структура программы

- Подключение библиотек и директив препроцессора
- Объявление глобальных переменных
- Тело программы

```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     cout << "Hello, world!";
6 }
```



# ЧАСТЬ 2. Переменные и типы данных



# Переменные



**Переменная** – именованная область памяти ЭВМ, в которой программа может хранить данные определенного типа (называемые значением переменной) и обращаться к этим данным, используя имя переменной

```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     int a=1, b=4, c;
6     c = a+b;
7     cout << "Сумма a+b = " << c;
8 }
9
```



# Переменные. Именование

Правило	Пример
1. Можно именовать используя много слов, но слитное написание. Никаких пробелов	bigRedButton, productOfNumbers, textOfMessage
2. Называть осознанно!	<del>a, b, c</del> --> firstTerm, messageArea, index, summa
3. Имя переменной может содержать только буквы латиницы, цифры и знаки _ и \$	button1, _LOCATION
4. Нельзя именовать переменные ключевыми словами C++	<del>int, String, char, if, for</del>
5. При обращении к переменной иметь ввиду регистр	Text и text – разные переменные!

# Типы данных



**Тип данных** – множество значений и операций над этими значениями.

В C++: bool, byte, short, int, long, double, float... итд

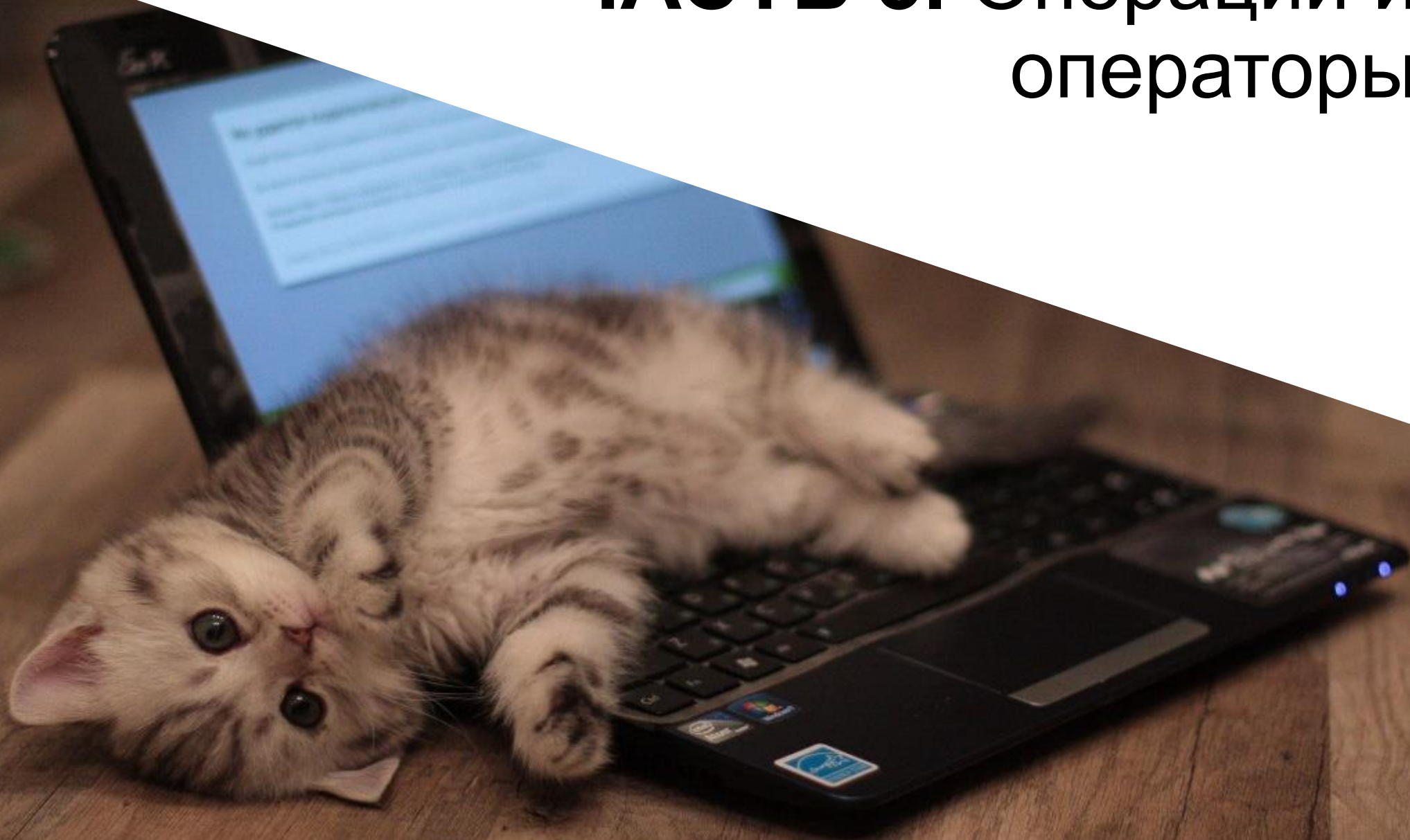
```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     int firstNumber = 1, secondNumber = 8, sumOfNumbers = 0;
6     sumOfNumbers = firstNumber + secondNumber;
7     cout << "Сумма чисел равна " << sumOfNumbers;
8 }
```

# Константы

**Константа** – (также как и переменная: именованная область памяти ЭВМ, в которой программа может хранить данные определенного типа и обращаться к этим данным, используя имя), но константа:

1. Не изменяется по мере исполнения программы. Если попытаться изменить константу, компилятор выдаст ошибку.
2. Объявляется с помощью ключевого слова **const**.

# ЧАСТЬ 3. Операции и операторы



# Операторы

**Оператор** (инструкция, команда) – наименьшая автономная часть языка программирования, способная выполняться самостоятельно и несущая самостоятельный смысл. Программа представляет собой последовательность операторов

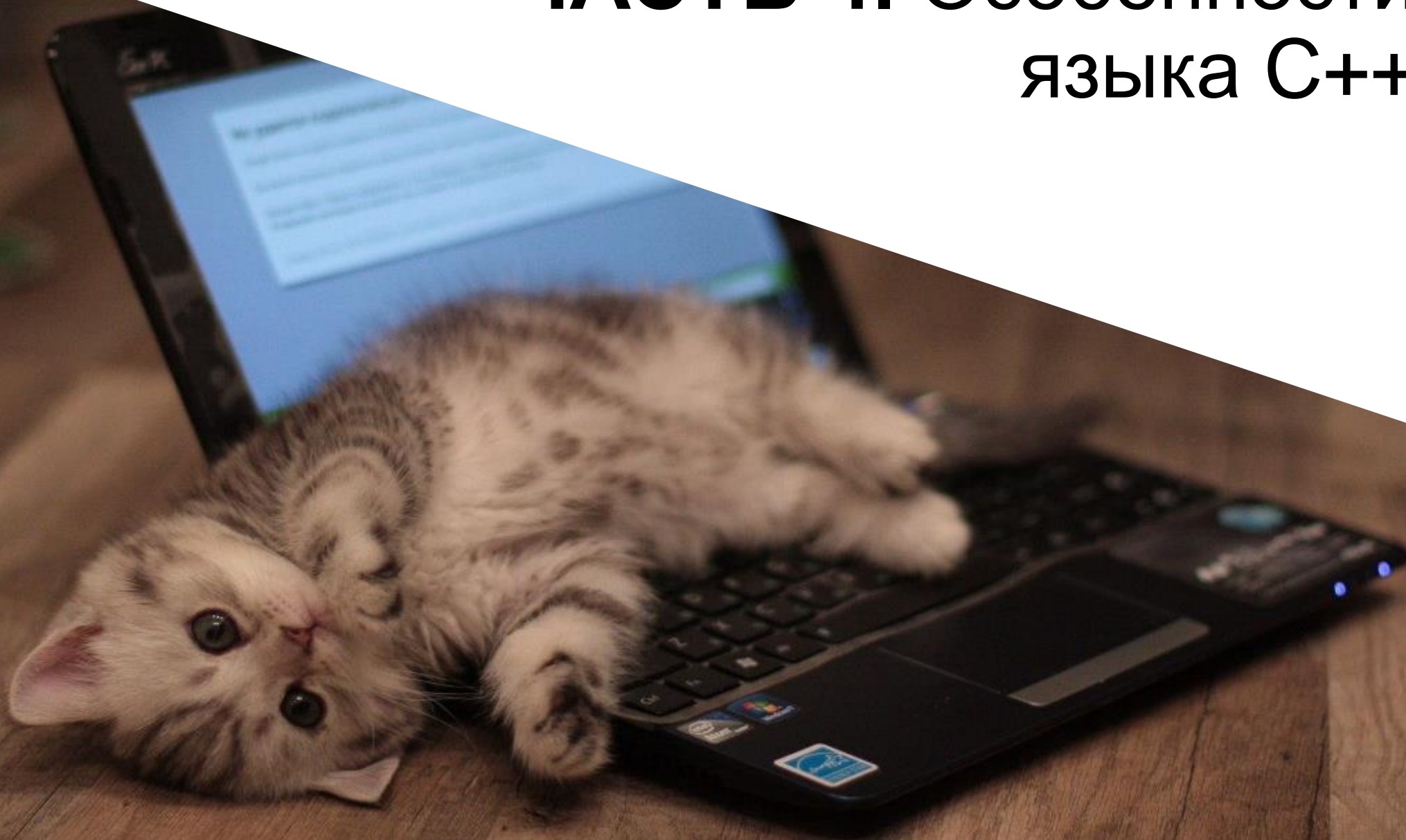
```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     int firstNumber = 1, secondNumber = 8, sumOfNumbers = 0;
6     sumOfNumbers = firstNumber + secondNumber;
7     cout << "Сумма чисел равна " << sumOfNumbers;
8 }
```



# Операторы в C++

Арифметические			
+	-	Сложение	Вычитание
*	/	Умножение	Деление
%		Остаток от деления	
++		Инкремент (увеличение на 1)	
--		Декремент (уменьшение на 1)	
Сравнения			
==	!=	Равно	Не равно
<	>	Меньше	Больше
<=	>=	Меньше или равно	Больше или равно
Логические			
&&		И	Или
!		Не	

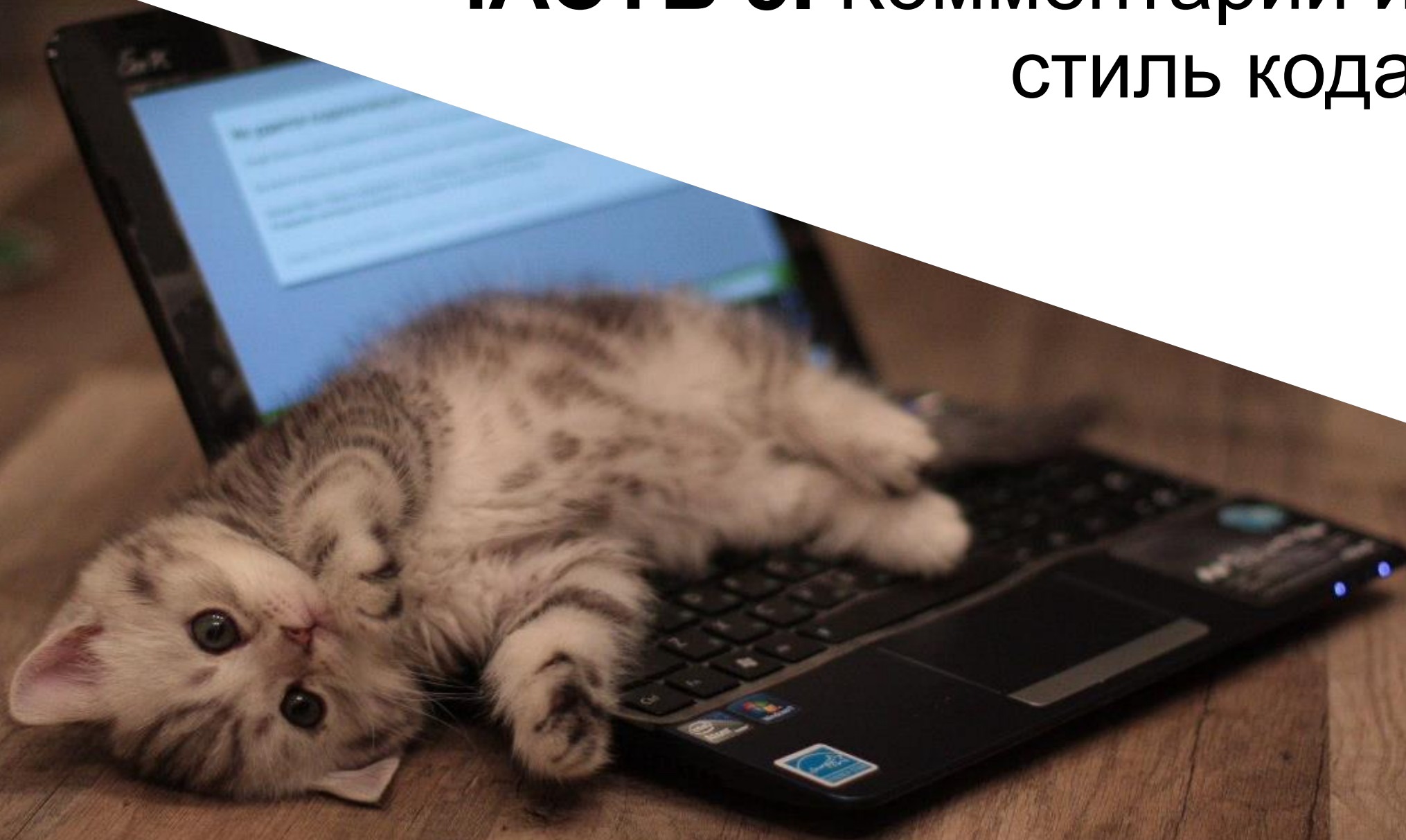
# ЧАСТЬ 4. Особенности языка C++



# Особенности C++

1. Язык C++ различает прописные и строчные буквы. Это означает, что имена всех функций и ключевые слова следует записывать в точности так, как они значатся в примерах и справочниках.
2. Каждая команда (оператор, строка) в языке C++ должна заканчиваться точкой с запятой.
3. C++ поддерживает реализацию ООП, и в то же время позволяет работать с данными на низком уровне.
4. Совместим с C, синтаксически схож с Java, C, C#.

# ЧАСТЬ 5. Комментарии и стиль кода



# Комментарии

**Комментарии** — это поясняющие надписи, которые используют программисты для улучшения понятности кода. При компиляции программы комментарии игнорируются.

```
1 // Подключаем библиотеки и директивы
2 #include <iostream>
3 using namespace std;
4
5 int main() { // Тут должно быть тело главной функции
6     // Объявляем переменные
7     int firstNumber = 1, secondNumber = 8, sumOfNumbers = 0;
8     sumOfNumbers = firstNumber + secondNumber;
9     // Выводим результат в консоль
10    cout << "Сумма чисел равна " << sumOfNumbers;
11 }
```



# Комментарии в C++



- `//` — если комментарий на одну строку
- `/* ... */` — комментарий на несколько строк

```
1  // Подключаем библиотеки и директивы
2  #include <iostream>
3  using namespace std;
4  /* Если этот код работает, его писала ФМА
5   * Если нет - не знаю кто его писал*/
6  int main() { // Тут должно быть тело главной функции
7              // Объявляем переменные
8              int firstNumber = 1, secondNumber = 8, sumOfNumbers = 0;
9              sumOfNumbers = firstNumber + secondNumber;
10             // Не знаю, идеален ли этот код, но пока пусть полежит тут
11             /* cout << "Сумма чисел равна " << sumOfNumbers; */
12 }
```

# Комментарии в C++

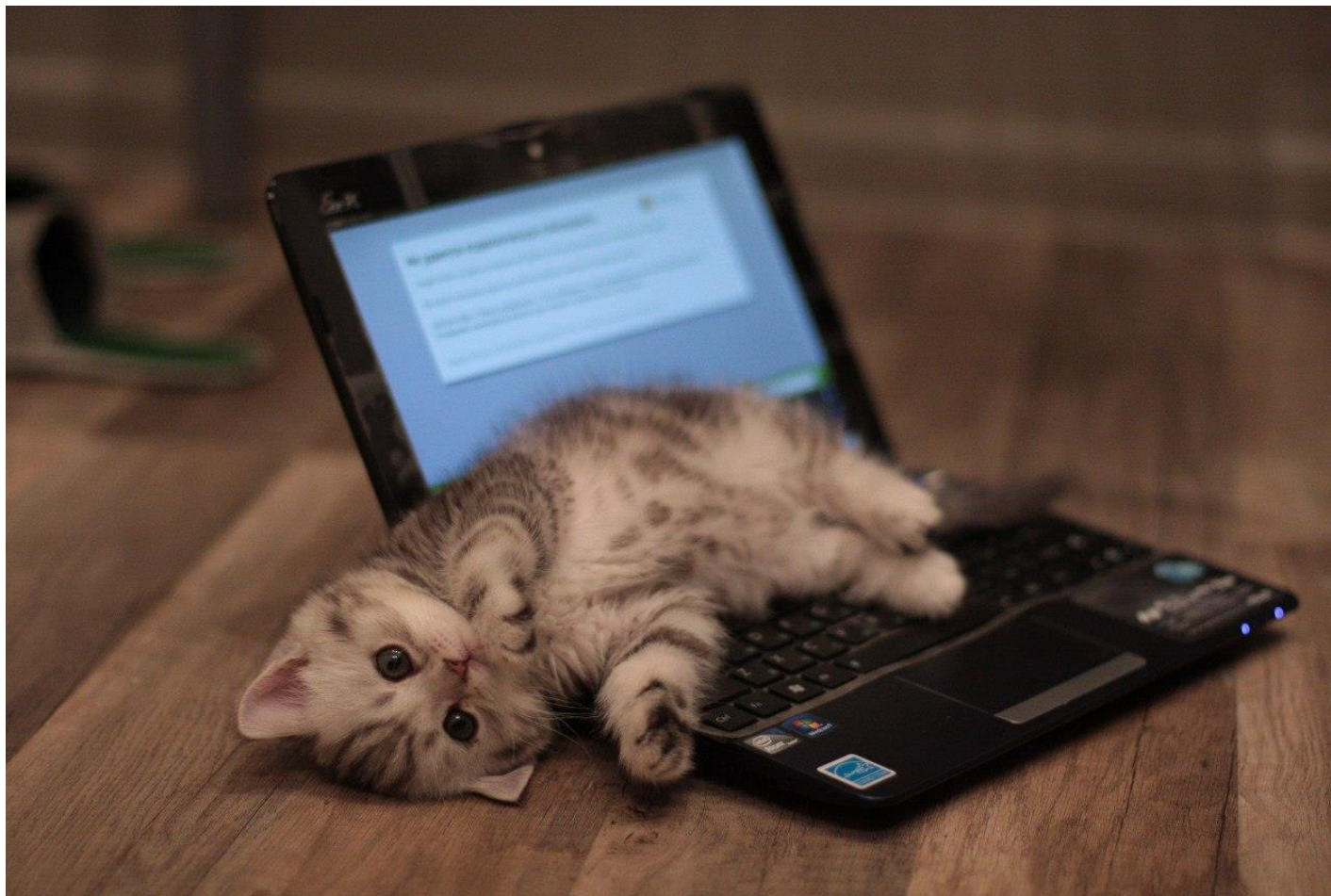
- `//` — если комментарий на одну строку
- `/* ... */` — комментарий на несколько строк

Комментировать можно:


- Подсказки и пояснения
- Сомнительный код

```
1 // Подключаем библиотеки и директивы
2 #include <iostream>
3 using namespace std;
4 /* Если этот код работает, его писала ФМА
5  * Если нет - не знаю кто его писал*/
6 int main() { // Тут должно быть тело главной функции
7     // Объявляем переменные
8     int firstNumber = 1, secondNumber = 8, sumOfNumbers = 0;
9     sumOfNumbers = firstNumber + secondNumber;
10    // Не знаю, идеален ли этот код, но пока пусть постоит тут
11    /* cout << "Сумма чисел равна " << sumOfNumbers; */
12 }
```

# Благодарю за внимание!



# Задания

1. Найти в презентации все слайды, отмеченные значком  (их 4).
2. Переписать, скомпилировать и запустить программы, код которых представлен на этих слайдах.
3. Видоизменить код (имена переменных, типы данных, операторы и текст комментариев) для получения собственных программ.

# Онлайн компиляторы

Задания можно выполнять в онлайн компиляторах.

Предлагаемые онлайн компиляторы:

1. Repl IT – <https://repl.it/languages/cpp>
2. Dcoder – <https://code.dcoder.tech/>
3. OnlineGDB – [https://www.onlinegdb.com/online\\_c++\\_compiler](https://www.onlinegdb.com/online_c++_compiler)
4. Jdoodle – <https://www.jdoodle.com/online-compiler-c++/>
5. IDEone – <https://ideone.com/l/cpp>
6. Programiz – <https://www.programiz.com/cpp-programming/online-compiler/>