

ПРОГРАММИРОВАНИЕ. Практика

Функции в C++



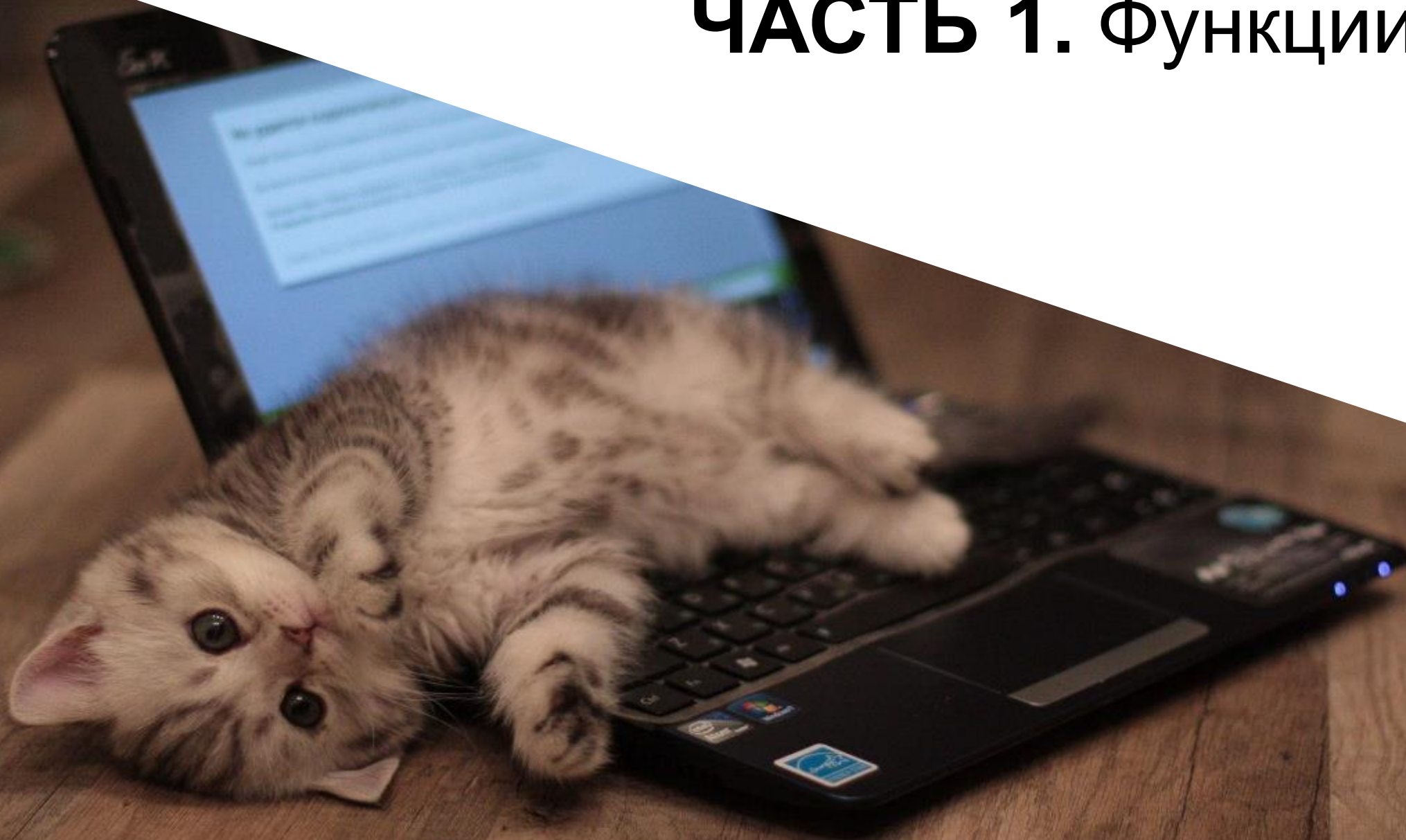
ФЕДОТЕНКО

Мария Александровна

 ma.fedotenko@mpgu.su

 [marusyafm](#)

ЧАСТЬ 1. Функции



Функция

Функция – поименованный фрагмент кода, реализованный с целью выполнения определенной последовательности операций и возможностью многократного использования в процессе выполнения программы



Преимущества функций

Использование функций имеет ряд преимуществ:

- **Сокращение объема кода.** Каждый новый вызов функции занимает всего 1 строку
- **При необходимости внесения изменений в логику программы достаточно изменить только тело функции,** без необходимости искать и изменять все ее вызовы



Знакомство с функциями

На самом деле вы уже знакомы с одной функцией.

Функция `main()` по умолчанию есть в каждой программе на C++.

Теперь вы научитесь использовать несколько функций в одной программе.

```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     cout << "Привет!";
6 }
7
```

Объявление функции

Для использования функции ее нужно **объявить**.

При объявлении функции указывается:

- **Возвращаемый тип данных** функции
- **Имя** функции
- **Параметры** функции

```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     cout << "Привет!";
6 }
7
```

Объявление функции

Для использования функции ее нужно **объявить**.

При объявлении функции указывается:

- **Возвращаемый тип данных функции**
- **Имя функции**
- **Параметры функции** (может не быть)

```
1 ▾ #include <iostream>
2   using namespace std;
3
4 ▾ int main() {
5     cout << "Привет!";
6 }
7
```

Объявление функции



Для использования функции ее нужно **объявить**.

При объявлении функции указывается:

- **Возвращаемый тип данных функции**
- **Имя функции**
- **Параметры функции**

```
1 ▾ #include <iostream>
2   using namespace std;
3   // Функция вычисления суммы
4 ▾ int summa(int firstNumber, int secondNumber) {
5     int sumOfNumbers = firstNumber + secondNumber;
6     cout << "Сумма чисел: " << sumOfNumbers << endl;
7     return sumOfNumbers;
8 }
```


Возвращаемый тип данных



Функции должны иметь **возвращаемый тип данных** и **возвращаемое значение**. При этом это должен быть один и тот же тип данных.

```
4 // Целочисленная функция
5 int somma(int firstNumber, int secondNumber){
6     int sumOfNumbers = firstNumber + secondNumber;
7     cout << "Сумма чисел: " << sumOfNumbers << endl;
8     // Возвращает целочисленное значение
9     return sumOfNumbers;
10 }
11
12 // Строковая функция
13 string text(string word){
14     word += "!";
15     // Возвращает строковое значение
16     return word;
17 }
```

Возвращаемый тип данных



Функции должны иметь **возвращаемый тип данных** и **возвращаемое значение**. При этом это должен быть один и тот же тип данных.

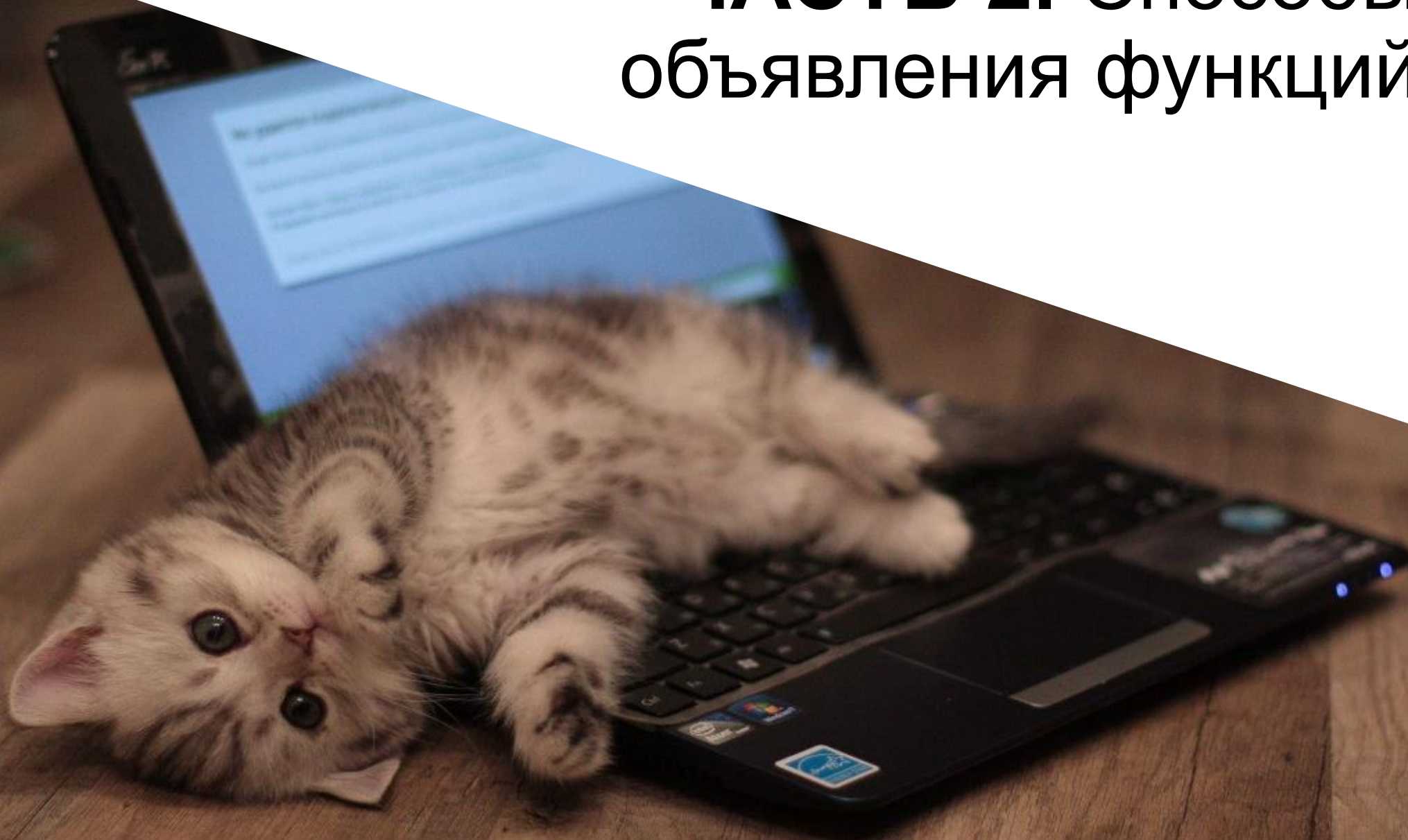
```
19 // Функция типа данных void (тип данных "отсутствие типа данных")
20 ▾ void message(){
21     cout << "Важное сообщение!";
22     // Ничего не возвращает
23 }
24
25 // Главная функция (целочисленная)
26 ▾ int main (){
27     cout << "Изучаем функции";
28     // main() - особенная функция (главная), ей не обязательно иметь возвращаемое значение
29     // но можно вернуть целочисленное - 0
30     return 0;
31 }
```

Вызов функции

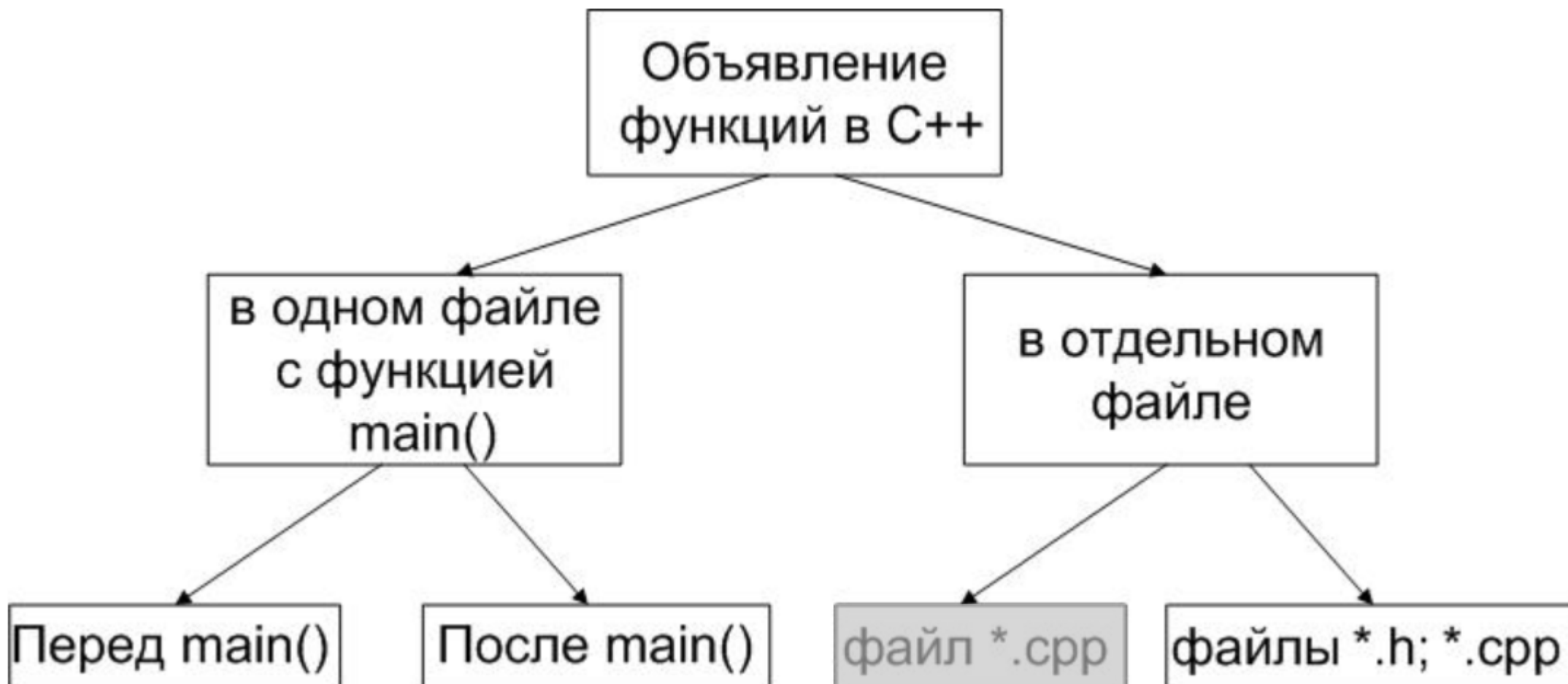


```
1 ▾ #include <iostream>
2  using namespace std;
3  // Функция вычисления суммы
4 ▾ int somma(int firstNumber, int secondNumber) {
5      int sumOfNumbers = firstNumber + secondNumber;
6      cout << "Сумма чисел: " << sumOfNumbers << endl;
7      return sumOfNumbers;
8  }
9  // Главная функция
10 ▾ int main() {
11      int number, anotherNumber;
12      cout << "Эта программа вычисляет сумму чисел" << endl;
13      cout << "Введите 2 целых числа:" << endl;
14      cin >> number >> anotherNumber;
15      somma(number, anotherNumber);
16      cout << "Введите еще 2 целых числа:" << endl;
17      cin >> number >> anotherNumber;
18      somma(number, anotherNumber);
19  }
```

ЧАСТЬ 2. Способы объявления функций



Объявление функции



Объявление перед main()



```
1 ▾ #include <iostream>
2  using namespace std;
3  // Функция вычисления суммы
4 ▾ int somma(int firstNumber, int secondNumber) {
5      int sumOfNumbers = firstNumber + secondNumber;
6      cout << "Сумма чисел: " << sumOfNumbers << endl;
7      return sumOfNumbers;
8  }
9  // Главная функция
10 ▾ int main() {
11      int number, anotherNumber;
12      cout << "Эта программа вычисляет сумму чисел" << endl;
13      cout << "Введите 2 целых числа:" << endl;
14      cin >> number >> anotherNumber;
15      somma(number, anotherNumber);
16      cout << "Введите еще 2 целых числа:" << endl;
17      cin >> number >> anotherNumber;
18      somma(number, anotherNumber);
19  }
```


Объявление после main()



```
1 ▾ #include <iostream>
2  using namespace std;
3  // Прототип функции вычисления суммы
4  int somma(int, int);
5  // Главная функция
6 ▾ int main() {
7      int number, anotherNumber;
8      cout << "Эта программа вычисляет сумму чисел" << endl;
9      cout << "Введите 2 целых числа:" << endl;
10     cin >> number >> anotherNumber;
11     somma(number, anotherNumber);
12     cout << "Введите еще 2 целых числа:" << endl;
13     cin >> number >> anotherNumber;
14     somma(number, anotherNumber);
15 }
16 // Функция вычисления суммы
17 ▾ int somma(int firstNumber, int secondNumber) {
18     int sumOfNumbers = firstNumber + secondNumber;
19     cout << "Сумма чисел: " << sumOfNumbers << endl;
20     return sumOfNumbers;
21 }
```

Объявление отдельным файлом

```
1 // Файл summa.h
2 #ifndef summa
3 #define summa
4 int summa(int, int); // прототип функции вычисления суммы
5 #endif
6
```

```
1 // Файл summa.cpp
2 #include "stdafx.h"
3 #include "summa.h"
4 int summa(int firstNumber, int secondNumber) {
5     sumOfNumbers = firstNumber + secondNumber;
6     cout << "Сумма чисел: " << sumOfNumbers << endl;
7     return sumOfNumbers;
8 }
```

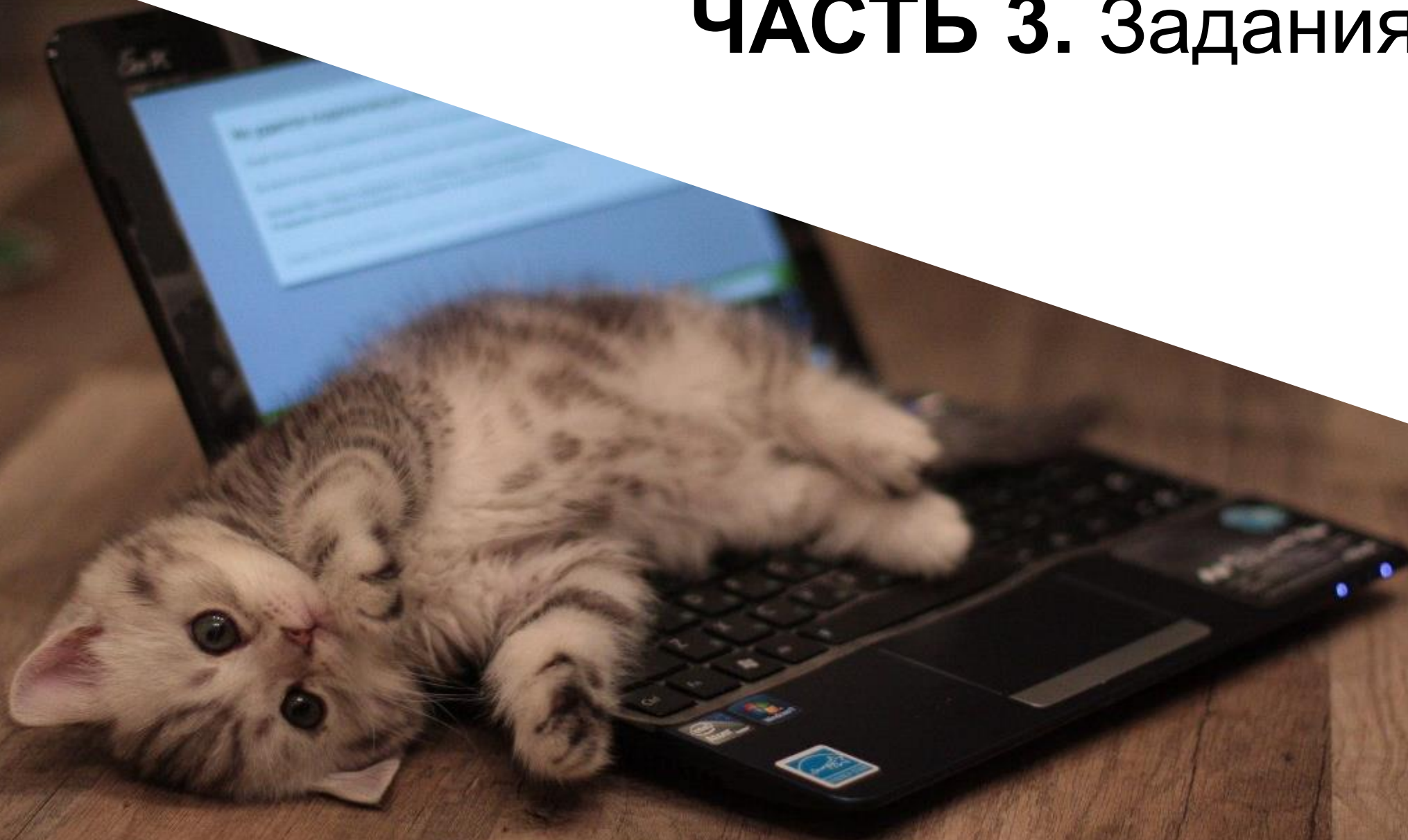
```
1 #include <iostream>
2 using namespace std;
3 // Подключение библиотек и файлов
4 #include "stdafx.h"
5 #include "summa.h"
6 // Главная функция
7 int main() {
8     cout << "Эта программа вычисляет сумму чисел" << endl;
9     cout << "Введите 2 целых числа:" << endl;
10    cin >> firstNumber >> secondNumber;
11    summa(firstNumber, secondNumber);
}
```

Глобальные переменные



```
1 ▾ #include <iostream>
2  using namespace std;
3  // Прототип функции вычисления суммы
4  int summa(int, int);
5  // Глобальные переменные
6  int firstNumber, secondNumber, sumOfNumbers;
7  // Главная функция
8 ▾ int main() {
9      cout << "Эта программа вычисляет сумму чисел" << endl;
10     cout << "Введите 2 целых числа:" << endl;
11     cin >> firstNumber >> secondNumber;
12     summa(firstNumber, secondNumber);
13     cout << "Введите еще 2 целых числа:" << endl;
14     cin >> firstNumber >> secondNumber;
15     summa(firstNumber, secondNumber);
16 }
17 // Функция вычисления суммы
18 ▾ int summa(int firstNumber, int secondNumber) {
19     sumOfNumbers = firstNumber + secondNumber;
20     cout << "Сумма чисел: " << sumOfNumbers << endl;
21     return sumOfNumbers;
22 }
```

ЧАСТЬ 3. Задания



Задание 1



Написать программу, вычисляющую произведение введенных пользователем чисел.

Условия:

1. Ввод пользователем чисел запрашивается в функции `main()`.
2. Вычисление произведения и вывод результата осуществляется в другой функции.
3. Другая функция должна вызываться из функции `main()` не менее 3х раз (перед этим пользователь каждый раз должен вводить новые числа).

Задание 2



Написать программу, которая просит пользователя задать с клавиатуры размер массива, заполняет его (псевдо)рандомными числами и выводит на экран.

Условия:

1. Запрос на ввод размерности массива реализуется в функции `main()`.
2. Заполнение и вывод на экран массива реализуется в другой функции, которая получает на вход значение размерности массива из функции `main()`.
3. Прodelать эту операцию для 3х массивов (то есть пользователь должен ввести 3 числа и вторая функция должна вызываться 3 раза).

Задание 3



Доработать программу из Задания 2 с учетом следующих условий:

1. Введенное пользователем число проверяется на соответствие условию «не меньше 5 и не больше 11» (если условие не выполняется – завершение работы программы).
2. В новой функции дополнительно посчитать и вывести количество элементов массива с четными значениями.

Благодарю за внимание!

