

Ravesli [Ravesli](#)

- [Уроки по C++](#)
- [OpenGL](#)
- [SFML](#)
- [Qt5](#)
- [RegExr](#)
- [Ассемблер](#)
- [Купить .PDF](#)



Глава №1. Итоговый тест

[Юрий](#) |

- [Уроки C++](#)

|

Обновлено 2 Сен 2020 |

51382

[↓](#) 68

Данные задания призваны улучшить запоминание пройденного материала и показать его применение на практике.

Оглавление:

1. [Задание №1](#)
2. [Задание №2](#)
3. [Задание №3](#)

Задание №1

Напишите однофайловую программу (с именем `main.cpp`), которая запрашивает у пользователя два целых числа, складывает их, а затем выводит результат. В программе должно быть 3 функции:

- функция `readNumber()`, которая запрашивает у пользователя целое число, а затем возвращает его в `main()`;
- функция `writeAnswer()`, которая выводит результат на экран. Функция должна быть без возвращаемого значения и иметь только один параметр;
- функция `main()`, которая соединяет всё и вся.

Подсказки:

- Для выполнения операции сложения не нужно создавать отдельную функцию (просто используйте оператор `+`).

→ Функцию `readNumber()` нужно вызывать дважды.

Ответ №1

main.cpp:

```
1  #include <iostream>
2
3  int readNumber()
4  {
5      std::cout << "Enter a number: ";
6      int x;
7      std::cin >> x;
8      return x;
9  }
10
11 void writeAnswer(int x)
12 {
13     std::cout << "The answer is " << x << std::endl;
14 }
15
16 int main()
17 {
18     int x = readNumber();
19     int y = readNumber();
20     writeAnswer(x + y); // передаем результат в функцию writeAnswer()
21     return 0;
22 }
```

Задание №2

Измените программу из задания №1 так, чтобы функции `readNumber()` и `writeAnswer()` находились в отдельном файле `io.cpp`. Используйте предварительные объявления для доступа к этим функциям с функции `main()`.

Подсказка: Если у вас возникли проблемы, убедитесь, что `io.cpp` правильно добавлен к вашему проекту и подключен к компиляции (см. [Урок №20. Многофайловые программы](#)).

Ответ №2

io.cpp:

```
1  #include <iostream>
2
3  int readNumber()
4  {
5      std::cout << "Enter a number: ";
6      int x;
7      std::cin >> x;
8      return x;
9  }
10
11 void writeAnswer(int x)
```

```
12 | {  
13 |     std::cout << "The answer is " << x << std::endl;  
14 | }
```

main.cpp:

```
1 | // Это предварительные объявления функций, которые находятся в файле io.cpp  
2 | int readNumber();  
3 | void writeAnswer(int x);  
4 |  
5 | int main()  
6 | {  
7 |     int x = readNumber();  
8 |     int y = readNumber();  
9 |     writeAnswer(x+y);  
10 |    return 0;  
11 | }
```

Задание №3

Измените программу из задания №2 так, чтобы она использовала заголовочный файл io.h для доступа к функциям (вместо использования предварительных объявлений). Убедитесь, что ваш заголовочный файл использует header guards.

Ответ №3

io.cpp:

```
1 | #include <iostream>  
2 |  
3 | int readNumber()  
4 | {  
5 |     std::cout << "Enter a number: ";  
6 |     int x;  
7 |     std::cin >> x;  
8 |     return x;  
9 | }  
10 |  
11 | void writeAnswer(int x)  
12 | {  
13 |     std::cout << "The answer is " << x << std::endl;  
14 | }
```

io.h:

```
1 | #ifndef IO_H  
2 | #define IO_H  
3 |  
4 | int readNumber();  
5 | void writeAnswer(int x);  
6 |  
7 | #endif
```

main.cpp:

```
1 #include "io.h"
2
3 int main()
4 {
5     int x = readNumber();
6     int y = readNumber();
7     writeAnswer(x+y);
8     return 0;
9 }
```

Оценить статью:



(479 оценок, среднее: 4,94 из 5)



[← Урок №27. Отладка программ: стек вызовов и отслеживание переменных](#)

[Урок №28. Инициализация, присваивание и объявление переменных →](#)



Комментариев: 68



1. Александр:

[31 августа 2020 в 22:54](#)

Спасибо!

Всё получилось.

Правда, пришлось кое-где возвращаться и исправлять ошибки (в основном — в настройках VS2019.

Был вопрос с выводом в консоли русского текста — тоже решил, покопавшись в комментариях к "Решениям распространённых проблем..." (решилось изменением шрифта в настройках консоли).

Рад, что влез в эти уроки.

Всем — добра))

[Ответить](#)



2. Pavel:

[20 июня 2020 в 11:13](#)

Получилось как то так:

main.cpp:

```
1 #include <iostream>
2 #include "source.h"
```

```
3 |
4 | int main() {
5 |     int a = readNumber();
6 |     int b = readNumber();
7 |     writeAnswer(add(a, b));
8 |     return 0;
9 | }
```

source.h:

```
1 | #ifndef ADD_H
2 | #define ADD_H
3 |
4 | int add(int a, int b);
5 | int readNumber();
6 | void writeAnswer(int a);
7 |
8 | #endif
```

source.cpp:

```
1 | #include <iostream>
2 |
3 | int add(int a, int b) {
4 |     return a + b;
5 | }
6 |
7 | int readNumber() {
8 |     std::cout << "Enter a number : " << std::endl;
9 |     int x = 0;
10 |    std::cin >> x;
11 |    return x;
12 | }
13 |
14 | void writeAnswer(int a) {
15 |     std::cout << "a + b = " << a << std::endl;
16 | }
```

[Ответить](#)



3. Вячеслав:

[10 мая 2020 в 10:25](#)

Добрый день,
у меня получился следующий код:

```
1 | #include <iostream>
2 |
3 | int readNumber()
4 | {
5 |     int x;
6 |     std::cout << "Введите число: " << std::endl;
```

```

7     std::cin >> x;
8     return x;
9 }
10
11 void writeAnswer(int a)
12 {
13     std::cout << "Сумма чисел: " << a << std::endl;
14 }
15
16 int main()
17 {
18     int a=readNumber();
19     int b= readNumber();
20     int sum = a+b;
21     writeAnswer(sum);
22 }

```

Ответить

1. *Александр:*
[12 мая 2020 в 22:31](#)

Вопрос по ответу- почему в параметре функции void writeAnswer() записана int x. ведь это один аргумент который этой функцией выводится на экран.ведь можно было в main() создать переменную допустим z которая равнялась x+y. И тогда в параметре writeAnswer() записали бы int z. и соответственно в самом теле .А так я не пойму- мы в параметре записали int x. И программа работает. Почему?

Ответить

4. *Андрей:*
[7 мая 2020 в 11:42](#)

Файл A.cpp:

```

1  #include <iostream>
2  using namespace std;
3
4  int readNumber()
5  {
6      cout << "Введи число:";
7      int a;
8      cin >> a;
9
10     return a;
11 }
12
13 void answer() {
14
15     int a(readNumber() + readNumber());
16     cout << "Ответ:" << a;

```

```
17 | }
```

Файл В.h:

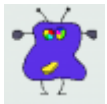
```
1 | int readNumber();
2 | void answer();
```

Файл main.cpp

```
1 | #include <iostream>
2 | #include "STD.h"
3 | using namespace std;
4 | int main()
5 | {
6 |     setlocale(LC_ALL, "ru");
7 |     answer();
8 | }
```

P.S мне компилятор помогает)
Я могу не писать хедер гварды)
так коротко и ясно)

[Ответить](#)



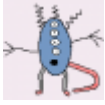
5. *Sagynysh:*

[30 апреля 2020 в 18:43](#)

Первое задание у меня получилось вот так 😊
Вроде всё корректно

```
1 | #include<iostream>
2 |
3 | using namespace std;
4 | int readNumber();
5 | void writeAnswer(int x);
6 |
7 | int readNumber()
8 | {
9 |     int x;
10 |     cout << "Enter a number: " ;
11 |     cin >> x ;
12 |     return 2*x;
13 | }
14 |
15 | void writeAnswer(int x)
16 | {
17 |     cout << "Result is: " << x << endl ;
18 | }
19 |
20 | int main()
21 | {
22 |     writeAnswer(readNumber());
```

```
23 |     return 0;
24 | }
```

[Ответить](#)1. *Andrey:*[2 мая 2020 в 20:24](#)

У вас не правильно так как вы умножаете целое число на два, вместо того, чтобы выводить сумму двух целых чисел. То есть пользователь может ввести $4 + 5 = 9$, а у вас будет $4 * 2 = 8$

[Ответить](#)6. *Хей.Лонг:*[5 апреля 2020 в 03:28](#)

А у меня получилась совсем новая концепция. Я не сразу понял, что сумму нужно передавать аргументом. Хотя учитывая ограниченный функционал этой тестовой работы, я не хочу считать мой вариант некорректным. По крайней мере оно скомпилировалось и выдаёт адекватный результат.

```
1 | #include <iostream>
2 |
3 | int readNumber() { // функция просит ввести число, и возвращает его коlobку.
4 |     int x = 0;
5 |     std::cout << "Введите число - ";
6 |     std::cin >> x;
7 |     return x;
8 | }
9 |
10 | void writeAnswer() { // функция выводит сумму двух вводимых чисел.
11 |     std::cout << "Сумма этого - " << readNumber() + readNumber() << std::endl;
12 | }
13 |
14 | int main()
15 | {
16 |     setlocale(LC_ALL, "Russian");
17 |     std::cout << "Программа суммирует два вводимых числа.\n";
18 |     writeAnswer();
19 |     return 0;
20 | }
```

[Ответить](#)7. *александр:*[20 ноября 2019 в 21:02](#)

немного непонятен момент: почему в нашем тесте тип возврата "void writeAnswer(int x)" мы не можем написать без инициализации (типа вот так — void writeAnswer()). а с "int" такого не

происходит. Пожалуйста объясните! Спасибо!

[Ответить](#)



1. Константин:

[27 декабря 2019 в 11:13](#)

(int x) — это такая ловушка, созданная в функции, для улавливания аргумента, передаваемого коллером — здесь значение будет схвачено и отхвачено согласно "Протоколам сионских мудрецов":-)шутка — обработано стейтментами, прописанными в колли (Caller -> Callee)

[Ответить](#)



8. A-Lex:

[16 ноября 2019 в 11:08](#)

(не судите строго за имена переменный — писал на работе, спешил)
Первое задание получилось както так:

```
1  #include <iostream>
2  using namespace std;    //чтобы постоянно не писать "std::"
3  int readNumber()
4  {
5      int x;
6      cout << "Enter number: ";
7      cin >> x;
8      return x;
9  }
10
11 void writeAnswer(int z)
12 {
13     cout << "result: " << z << endl;
14 }
15
16 int main()
17 {
18     writeAnswer(readNumber() + readNumber());
19     return 0;
20 }
```

[Ответить](#)



9. Андрей:

[4 ноября 2019 в 16:29](#)

По другому не смог решить первое задание 😊

```
1  #include <iostream>
2
3  int readNumber1()
4  {
```

```
5     int x;
6     std::cin >> x;
7     return x;
8 }
9
10 int readNumber2()
11 {
12     int y;
13     std::cin >> y;
14     return y;
15 }
16
17 void writeAnswer(int x, int y)
18 {
19     std::cout << x + y << std::endl;
20 }
21
22 int main()
23 {
24     int x = readNumber1();
25     int y = readNumber2();
26     int z = x + y;
27     std::cout << z << std::endl;
28 }
```

[Ответить](#)

1. Юра:

[18 ноября 2019 в 19:20](#)

Андрей, попробуйте применить функцию readNumber1() для ввода значения x и значения y в int main(). Получится, что readNumber2() не нужна.

[Ответить](#)

10. Алексей:

[24 сентября 2019 в 22:55](#)

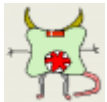
```
1 // Вот мой улучшенный пример программы для Задания №1
2
3 #include <iostream>
4
5 int a = 0;
6
7 int readNumber()
8 {
9     a = a + 1;
10    if (a == 1)
11        std::cout << "Enter the FIRST positive integer: ";
12    if (a == 2)
13        std::cout << "Enter the SECOND positive integer: ";
```

```
14     int x=0;
15     std::cin >> x;
16     if (x <= 0 || x > 100000000000000000000) std::cout << "THIS IS AN INCORRECT
17     return x;
18 }
19
20 void writeAnswer(int x)
21 {
22     std::cout << "The answer is numbers ONE+TWO=" << x << std::endl;
23 }
24
25 int main()
26 {
27     int x = readNumber();
28     if (x <= 0) return 0;
29     std::cout << "A number ONE is: " << x << std::endl;
30     int y = readNumber();
31     if (y <= 0) return 0;
32     std::cout << "A number TWO is: " << y << std::endl;
33     writeAnswer(x + y);
34     return 0;
35 }
```

[ОТВЕТИТЬ](#)

1. Андрей:
[4 ноября 2019 в 17:05](#)

Жестка

[ОТВЕТИТЬ](#)

11. Steve:
[12 сентября 2019 в 16:45](#)

```
1 #include <iostream>
2
3 using namespace std;
4
5 int readNumber()
6 {
7     cout << "Введите число: ";
8     int a;
9     cin >> a;
10    return a;
11 }
12
13 void writeAnswer(int total)
14 {
15     cout << "Сумма чисел равна: " << total;
16 }
```

```
17
18 int main()
19 {
20     int x = readNumber();
21     int y = readNumber();
22
23     int sum = x + y;
24
25     writeAnswer(sum);
26
27     return 0;
28 }
```

Качественные и бесплатные уроки. Написано все доходчиво, простым языком с детальным разъяснением, в отличие от некоторых книг, якобы для "начинающих". Огромное спасибо за ваш труд. Ставлю 5 звезд под каждым уроком 😊

[Ответить](#)



1. Юрий:

[15 сентября 2019 в 11:18](#)

Спасибо! Очень приятно)

[Ответить](#)



12. Максим:

[24 августа 2019 в 17:43](#)

```
1 #include <iostream>
2
3 int readNumber()
4 {
5     int x;
6     std::cout << "vvedi 4islo:  ";
7     std::cin >> x;
8     return x;
9 }
10
11 int writeAnswer(int x)
12 {
13     std::cout << x << std::endl;
14     return x; // без этой команды не хотело компилироваться
15 }
16
17 int main()
18 {
19     int x = readNumber();
20     int y = readNumber();
21     writeAnswer(x + y);
22     return 0;
23 }
```

23 | }

[Ответить](#)1. *Максим:*[25 августа 2019 в 10:10](#)

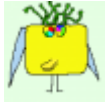
Я короче понял. Это у вас ошибка в ответе на первое задание с void и int

[Ответить](#)1. *Денис:*[24 января 2020 в 08:00](#)

Это у вас ошибка.

Никакого return x там не надо.

Функцию требовалось объявить void.

[Ответить](#)13. *Максим:*[16 июля 2019 в 12:11](#)

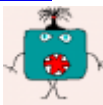
Спасибо за уроки и потраченное время на их создание. Пока что в первой главе всё понятно. Я пытался выучить c++ до этого по другим урокам, но там почти ничего не понятно.

[Ответить](#)1. *Юрий:*[16 июля 2019 в 16:04](#)

Пожалуйста))

[Ответить](#)14. *zevs:*[6 июля 2019 в 22:48](#)

Что то не понимаю, если количество аргументов должно совпадать с количеством параметров, то почему writeAnswer с двумя аргументами в main компилируется, при том, что в определении функции writeAnswer, мы указали один параметр. Те параметр один а аргументов два.

[Ответить](#)1. *ShnokDog:*[7 июля 2019 в 15:08](#)

Нет, вы передаёте параметр в виде суммы то есть сперва x прибавится к у потом полученное число передастся в виде одного аргумента!

[Ответить](#)

1. zevs:

[7 июля 2019 в 18:23](#)

В таком случае каким образом сумма выражения $x + y$ из `main()`, присваивается переменной(?) `X` в функции `writeAnswer()`? И вообще является ли `X` переменной, т.к в функции `writeAnswer()` мы ее нигде не определяем. Так каким образом компилятор узнает что результат вычисления выражения $x+y$, надо присваивать `X` (что это вообще?)

Или же `int x`, которую мы указываем в параметре функции является неинициализированной переменной, и результат вычисления автоматически присваивается ей? В таком я просто этого не знал, т.к раньше об этом в этом курсе не писалось.

[Ответить](#)

1. zevs:

[7 июля 2019 в 18:39](#)

Я просто как прочитал задание сразу понял как это реализовать через возвращаемую функцию,

```
1 int writeAnswer(int num1, int num2) {
2     int result = num1 + num2;
3     cout << "The answer is: " << result;
4     return num1 + num2;
5 }
```

И соответственно с двумя аргументами у вызываемой функции из `main()`

```
1 int main()
2 {
3     int num1 = readNumber();
4     int num2 = readNumber();
5     writeAnswer(num1, num2);
6     return 0;
7 }
```

Конечно, по условию задачи это будет неверным решением, но я просто не знал что можно проводить вычисление в месте для аргументов `()`, и автоматически передавать результат в неинициализированную переменную, которая к тому же определена в месте для параметров.

Возможно недостаточно внимательно читал уроки, и такая возможность там указывалась.

P.s вставил код, и не знаю как он отформатируется.



15. Алексей:

[24 июня 2019 в 16:08](#)

Немного тупил в первом задании, но все сразу.

Единственное — сам делаю с помощью vim в Убунту, мне надо включить в файл .h io.cpp.

Сам код:

```
1 #include "io.cpp"
2 #ifndef IO_H
3 #define IO_H
4 #endif
```

По ответу у меня ошибка.

```
/tmp/cccYR0a.o: In function main':
main.cpp:(.text+0x9): undefined reference to readNumber()
main.cpp:(.text+0x11): undefined reference to readNumber()'
main.cpp:(.text+0x23): undefined reference to writeAnswer(int)'
collect2: error: ld returned 1 exit status
```

[Ответить](#)



1. *Алексей:*

[4 июля 2019 в 11:10](#)

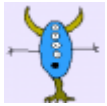
Замечание по 3му тесту.

Я уже за 37м уроком, но повтор не помешает. Тренировка, как-никак.

Так вот — я один раз объявил x для пустотной функции и этого достаточно.

Компилился, считается, проблем нет.

[Ответить](#)



16. *zashiki:*

[6 июня 2019 в 18:32](#)

Здравствуйте, насчет 1го вопроса: у меня вариант ниже, результат правильный, НО выходит такое:

введите число: 8

введите число: 7

сумма: 7+8=15

Почему числа в сумме меняются местами, никак не могу сообразить?

Сам код:

```
1 #include<iostream>
2
3 int readNumber()
4 {
5     std::cout << "введите число: ";
6
7     int a;
8     std::cin >> a;
```

```
9     return a;
10 }
11 void writeAnswer(int a, int b)
12 {
13     std::cout << "\nсумма: "<<a<<" + "<<b<<" = "<<a + b;
14 }
15 int main()
16 {
17     setlocale(0, "");
18
19
20     writeAnswer(readNumber(), readNumber());
21
22     std::cin.clear();
23     std::cin.ignore(32767, '\n');
24     std::cin.get();
25     return 0;
26 }
```

спасибо

[Ответить](#)



1. *zashiki:*

[6 июня 2019 в 18:40](#)

а, проглядела, что writeAnswer должна быть с одним параметром. Но все равно остается вопрос.

[Ответить](#)

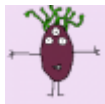


1. *Nikita:*

[6 июня 2019 в 22:47](#)

Аналогичный фейл с моментом что переменные меняются местами. В операциях суммирования и умножения это не проблема. А вот разницы и деления...

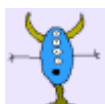
[Ответить](#)



1. *Алексей:*

[7 июня 2019 в 07:16](#)

У вас в строка "writeAnswer(readNumber(), readNumber());" компилятор не читает справа налево, а вначале запрашивает второй параметр потом первый, но вы этого понять не можете поскольку функция одна и та-же.



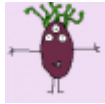
2. *zashiki:*

[7 июня 2019 в 10:20](#)

Алексей, извиняюсь за свою тупость, но:

1) если он запрашивает сначала 2й параметр, то почему выводит сначала 1 параметр, потом 2й, а потом уже складывает 2й+1й?

2) если разные функции в параметрах `writeAnswer(g1(), g2())`, то получается та же фигня,
НО если в этих функциях возвращается сразу число, к примеру, `a=1` или `return 1`, то все норм...



3. *Алексей:*
[7 июня 2019 в 21:51](#)

Объясняю (хотя каюсь все еще не запускал компилятор и все только рассуждения как должно быть) вы пишете:

```
"writeAnswer(readNumber(), readNumber());"
```

вводите 8 затем 7 и поскольку вначале запускается вторая функция то это будет тоже что:

`"writeAnswer(7, 8);"` — первая вызвалась функция `readNumber()` во втором параметре ! если хотите убедиться надо сделать не только функции с разным именем, но и с разным текстом запроса "Введите число 1" и "Введите число 2". Сама функция `writeAnswer(7, 8)` параметры не переставляет у нее все четко расписано как есть "a" есть "b" и выводит все правильно как в ней записано `7+8`, ошибка в том что вы думаете что первое число которое вы вводите оно подставляется в функцию первым, а на самом деле вы вводите 2-рой параметр функции.



17. *Анастасия:*
[10 мая 2019 в 18:32](#)

У меня всё получилось сделать ровно за 20 минут. Смущает, что нельзя прописать `#include <iostream>` только в главном файле (в одном месте), странно, что его нужно прописывать в каждом файле с функциями, которые могут его использовать.

[Ответить](#)



18. *Илья:*
[26 февраля 2019 в 19:22](#)

Здравствуйте!

Спасибо за Ваши труды!

У меня к Вам такой вопрос (может быть я где то упустил этот момент, если так, прошу просто скинуть ссылку=): у Вас в программе в исполнительных файлах обращение к `<iostream>` то есть, то нет, в итоге не понятно, когда есть необходимость подключать данную библиотеку, а когда нет? Не понимаю от слова совсем=)

Я создал `main` файл, в нем вбил `#include "pch.h"` и `#include <iostream>`, потом создал (по аналогии с вашим примером) исполнительный файл ю уже без использования данных директив, в итоге получил сбой ещё на этапе компиляции. добавил `#include "pch.h"` и `#include <iostream>` — всё заработало. попробовал убрать только `#include <iostream>` — программа продолжила стабильно работать. т.е. проблема именно в отсутствии `#include "pch.h"`?

Но, если я правильно понял статью, то когда мы создаем исполнительный файл, `main` остаётся

своего рода главным файлом, обращаясь лишь к информации в исполнительном, соответственно зачем ему директивы в исполнялке? Он вроде как должен читать код другого файла, используя свои директивы?

И про заголовочные файлы тоже немножко путаница в голове.

В общем, про многофайловые программы есть множество вопросов, был бы премного благодарен за ответы. И, если честно, думаю, Ваша аудитория была бы Вам тоже очень благодарна, если бы Вы сделали отдельный урок или статью о том, как заголовочные и исполнительные файлы взаимодействуют между собой. Ну т.е. подробно: какой файл к какому обращается, как они зависят друг от друга, можно ли с помощью заголовочного файла подключить библиотеки к исполнительным, или может это можно сделать с помощью другого файла. Было бы очень интересно понять этот процесс с точки зрения железа — как оно обрабатывает эти файлы.

Извините, если комментарий получился очень большим=) Просто очень хочется разобраться=)

[Ответить](#)



1. *Константин:*
[10 марта 2019 в 11:54](#)

Илья, ты объявления, стоя на остановках, читал? Вот заголовочный файл (с расширением .h) это то же объявление с остановки. В нём прописано нечто тебя интересующее и доступ к нему. По прописанному ты устанавливаешь доступ и получаешь интересующее тебя. Переходишь к следующему объявлению и всё повторяешь. Некоторые объявления написаны от руки, поэтому они в "таких" кавычках, некоторые напечатаны в типографии поэтому они в <таких> кавычках. Например, вызываешь функцию `girlByCall()`. В объявлении указано: `love girlByCall(long long double siski, char pussy);` `love` — это тип возвращаемого значения (т. е. ты от неё получаешь любовь), `long long double siski & char pussy` — это её параметры, в которые ты передаёшь свои аргументы — денежные знаки. `A void` — это например собственные дети — ничего тебе не возвращают, но аргументы ты им передаёшь. Ещё вариант — вызов например "Скорая Медицинская помощь": `help ambulance();`, которая хранится в библиотеке предварительно скомпилированных заголовков `<call911>` и приклеена к столбу (как и все другие заголовочные файлы) клеим `#include`. Ну и последний вариант: `void commentaryIntoInternet();` — ты не передаёшь туда никаких денежных значений и не ждёшь оттуда никаких поступлений. Просто комментарий живёт своей жизнью и производит какое-то воздействие на читателя... Я так себе это всё представляю.

[Ответить](#)



19. *Эрик:*
[5 февраля 2019 в 13:57](#)

Спасибо за уроки, сделал с первого раза. Была проблема с директивами и препроцессором в этой главе, пришлось обращаться к сторонним видео урокам, чтобы въехать в суть. Но в основном все легко усваивается. Спс)

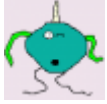
[Ответить](#)



20. *nickatin:*
[28 января 2019 в 12:52](#)

Получилось даже без ошибок. И все благодаря очень подробным урокам)

[Ответить](#)



21. *Antgon:*

[18 января 2019 в 19:51](#)

Хотел немного усложнить задание и добиться, чтобы в строке вывода был не просто результат сложения, а, например, так:

Итого: $5 + 6 = 11$

При написании в `main` строки

```
1 | std::cout<<"ИТОГО  " << x << "+" << y << "=" << x + y << std::endl;
```

все показывает как мне нужно, но при попытке написать код для функции `void writeAnswer` все идет кувырком. Перепробовал кучу вариантов, но без результата.

Пожалуйста, подскажите, как сделать правильный код для функции, чтобы при компиляции появлялся именно такой результат!

[Ответить](#)



1. *Константин:*

[19 января 2019 в 23:54](#)

Про побочные эффекты читал? Делай минимум действий в каждом стейт-менте — тогда и кувырки в прог-е меньше отжигать будут.

[Ответить](#)



2. *Alexey:*

[25 января 2019 в 14:53](#)

Скорее всего из-за того, что вы отправляете на в функцию печати всего одно значение, а хотите использовать все 3. Измените функцию так, чтобы она получала все 3 значения.

[Ответить](#)



22. *GAS:*

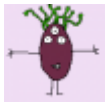
[11 декабря 2018 в 11:23](#)

```
1 | #include <iostream>
2 |
3 | using namespace std;
4 |
5 | int readNumber();
6 | void writeAnswer();
7 |
8 | int main()
9 | {
10 |     setlocale(0, "");
11 | ;
```

```
12     writeAnswer();
13
14     system("pause");
15     return 0;
16 }
17
18 int readNumber()
19 {
20     int a;
21     int b;
22
23     cout << "Введите два целых числа: ";
24     cin >> a >> b;
25
26     return a+b;
27 }
28
29 void writeAnswer()
30 {
31     cout << "\nРезультат: " << readNumber() << endl;
32 }
```

[Ответить](#)23. *Лёха:*[7 декабря 2018 в 16:03](#)

То чувство, когда все на изи этот тест пишут, а я ни одного задания не сделал, половину кода не понимаю. Придётся заново всё перечитывать, пока не начну понимать 😞

[Ответить](#)24. *ufo:*[29 октября 2018 в 12:37](#)

Пример хороший но не логичнее ли будет вписать `#include <iostream>` в `ю.h` (а также все нужные инклюды которые могут потребоваться для других функций).

После чего в `ю.cpp` и в `main.cpp` в инклюдах надо писать только `#include "ю.h"` (вместо дублирования инклюдов а их может быть много, тем более у нас хеадергвард прописан) в результате например `std::cout<<"ALL OK";` в конце `main` будет работать.

[Ответить](#)25. *Руслан:*[23 октября 2018 в 15:54](#)

```
1 #include <iostream>
2
3 using namespace std;
4
5 int readNumber()
```

```

6   {
7   cout<<"Vvedite celoe choslo "<<endl;
8   int num1;
9   cin >> num1;
10  return num1;
11  }
12  int summNum(int x, int y){
13  return x+y;
14  }
15  void writeAnswer(int result){
16  cout<<"Resultat "<<result<<endl;
17
18  }
19  int main()
20  { int num1=readNumber();
21    int num2=readNumber();
22    int result=summNum(num1,num2);
23    writeAnswer(result);
24
25    return 0;
26  }

```

а таким способом будет правильно?

[Ответить](#)



1. *Steve Dekart:*
[21 июня 2019 в 18:58](#)

Нет, там говорится использовать только 3 функции, а не 4.

[Ответить](#)



26. *Andrey:*
[11 октября 2018 в 13:32](#)

Решил «тупо» переписать ответы на тесты, что бы набить руку.(Итоговый тест 1). Первое задание откомпилировалось без проблем. А второе из 2-х файлов никак не компилируется. И в первом и во втором случае выдаёт ошибку 1010 и советуете добавить `#include 'stdafx.h'`, но я точно копирую ответ. В чем прикол? Я полный «чайник», но тема зашла, хочется дойти до сути.

[Ответить](#)



27. *Рома:*
[21 сентября 2018 в 00:21](#)

Здравствуйте, в 3 задании сделал точно также как и у вас в ответе, но мне выдает две ошибки :

- 1) идентификатор readNumber() не определён
 - 2) идентификатор writeAnswer() не определён
- в чём может заключаться проблема?

Проверял уже много раз код один в один с вашим .

[Ответить](#)1. *Алексей:*[25 сентября 2018 в 09:42](#)

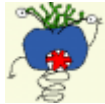
Тут 2 варианта. Либо в `main.cpp` не прописан файл заголовка (`#include "io.h"`), либо в файле заголовка не объявлены указанные функции. Подробнее здесь: Урок 21. Заголовочные файлы.

[Ответить](#)2. *Эрик:*[26 сентября 2018 в 20:51](#)

Подключение `io.h` идет до подключения `<iostream>`?

[Ответить](#)1. *Рома:*[27 сентября 2018 в 00:01](#)

Эрик, спасибо огромное, у меня в принципе `#include <iostream>` не был подключен в этом была вся проблема, не понимаю как я упустил это, ещё раз огромное спасибо)))

[Ответить](#)2. *Алексей:*[28 сентября 2018 в 10:37](#)

Файл заголовка можно подключить и до и после `jostream`.

[Ответить](#)28. *Tahir:*[19 августа 2018 в 12:58](#)

Отличный тест. Без лишней теории и охватывает все основные аспекты Главы 1. Спасибо за Ваш труд!

[Ответить](#)1. *Юрий:*[19 августа 2018 в 13:01](#)

Спасибо, что читаете 😊

[Ответить](#)



29. Даниил:

[11 августа 2018 в 18:28](#)

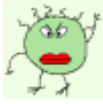
Здравствуйте , такой вопрос)

Во второй функции мы вводим одну переменную x

А в функции main() мы выводим эту функцию и (x+y)....

Уточните пожалуйста что здесь и как)

[Ответить](#)



1. Александр:

[12 августа 2018 в 01:24](#)

А что там не понятного? Мы вызываем функцию writeAnswer() А "x+y" — это параметр этой функции, который равен сумме x и y, и вот это значение присваивается переменной x в функции writeAnswer() Просто нужно внимательно читать уроки. У каждой функции своя переменная с именем "X" (в каждом городе есть своя улица Ленина)

[Ответить](#)



2. Алексей:

[20 сентября 2018 в 13:06](#)

Все просто. Каждая из представленных функций "не знает", что одноименные переменные есть в других функциях.

Подробнее тут: Урок 15. Локальная область видимости.

[Ответить](#)



30. Алексей:

[24 июня 2018 в 13:37](#)

Есть недочет. После каждого урока надо шпук по 10 практических заданий для закрепления материала. Все получилось с первого раза! Но практики очень не хватает.

[Ответить](#)



1. Максим:

[17 июля 2018 в 17:05](#)

Согласен, это уже будет полировка, а так все хорошо.

[Ответить](#)



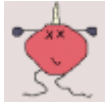
2. Константин:

[28 июля 2018 в 10:17](#)

Да-да, полностью поддерживаю Алексея: Юрий, пожалуйста, прибавьте контрольных заданий, а то как-то мало связей между нейронами образуется у меня в голове от одного

задания:-(

[Ответить](#)



31. *Илья:*

[17 июня 2018 в 12:05](#)

у меня в 1задании 39 ошибок, придется юдить сначала

[Ответить](#)



32. *Игорь...:*

[15 июня 2018 в 19:49](#)

Прикольно, первое задание выполнил так же как и в ответе, кроме...

```
1 int main()
2 {
3     writeAnswer(readNumber() + readNumber());
4     return 0;
5 }
```

Работает... Но правильно ли так?

[Ответить](#)



1. *Юрий:*

[15 июня 2018 в 20:27](#)

Здесь создаются анонимные объекты (об этом есть отдельный урок). В общем, для этого примера такой ответ тоже подойдет. Но если бы в дальнейшем нужно было еще что-либо выполнить с этими числами, то удобнее было бы через переменные.

[Ответить](#)



1. *Nikita:*

[14 мая 2019 в 15:48](#)

Я когда сделал вот так как на скрине (похоже), у меня почему то 1я и 2е число меняются местами. Например деление, происходит 2ого числа на 1е, и так и не понятно почему(

[Ответить](#)



33. *Diana:*

[15 июня 2018 в 08:27](#)

В итоге все получилось, не без ошибок, конечно, но получилось) очень рада) спасибо за Ваш труд!)

[Ответить](#)



1. *Юрий:*
[15 июня 2018 в 11:01](#)

Пожалуйста 😊

[Ответить](#)



34. *Дима:*
[6 июня 2018 в 15:41](#)

Хм....урок без коментов.
У всех все просто получилось?!)

[Ответить](#)

Добавить комментарий

Ваш E-mail не будет опубликован. Обязательные поля помечены *

Имя *

Email *

Комментарий

☐ Сохранить моё Имя и E-mail. Видеть комментарии, отправленные на модерацию

☐ Получать уведомления о новых комментариях по электронной почте. Вы можете [подписаться](#) без комментирования.





[TELEGRAM](#)  [КАНАЛ](#)



[ПАБЛИК](#) 

ТОП СТАТЬИ

-  [Словарь программиста. Сленг, который должен знать каждый кодер](#)

-  [Урок №1. Введение в программирование](#)
-  [70+ бесплатных ресурсов для изучения программирования](#)
-  [Урок №1: Введение в создание игры «Same Game»](#)
-  [Урок №4. Установка IDE \(Интегрированной Среды Разработки\)](#)

- [Ravesli](#)
- - [О проекте](#) -
- - [Пользовательское Соглашение](#) -
- - [Все статьи](#) -
- Copyright © 2015 - 2020