

Міністерство освіти і науки України
Харківський національний автомобільно-дорожній університет
Кафедра інформаційних технологій та мехатроніки

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт з дисципліни
“Інформаційні технології”

“Програмування на мові C++ у середовищі Microsoft Visual Studio 2010”

для студентів напрямку підготовки 6.050702 Електромеханіка

Розроблено та надруковано доц. Симбірським Г.Д.

Харків, 2018

Лабораторная работа № 1

ИССЛЕДОВАНИЕ ИНТЕРФЕЙСА ПРОГРАММЫ VISUAL STUDIO 2010. РАЗРАБОТКА ПРОГРАММ С ЛИНЕЙНОЙ СТРУКТУРОЙ В СРЕДЕ VISUAL C++ 2010. ИССЛЕДОВАНИЕ ПРОЦЕДУР ВВОДА И ВЫВОДА ДАННЫХ В СРЕДЕ VISUAL C++ 2010.

- Цель работы:**
1. Исследование интерфейса среды Visual Studio 2010;
 2. Приобретение навыков алгоритмизации задач;
 3. Исследование процессов ввода и вывода данных в Visual C++ 2010.
 4. Изучение приемов создания и отладки консольных проектов и программ с линейной структурой в среде программирования Visual C++ 2010;

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1. Общие сведения о программе Visual Studio 2010

Visual Studio 2010 — это полный набор средств для разработки программ и приложений для компьютеров, мобильных устройств и др. с использованием различных языков программирования. Среда разработки Visual Studio 2010 предназначена для написания программ и приложений на языках .NET, HTML, JavaScript, C# и C++. Возможности этого программного инструмента очень велики. Поэтому он получил название **ИСП - интегрированная среда разработки** (Integrated Development Environment – **IDE**). Не выходя за рамки этой среды, можно решать следующие задачи:

1. Формировать заготовки приложений без написания текстов программ;
2. Просматривать проекты несколькими разными способами;
3. Редактировать файлы заголовков и текстов программ;
4. Формировать визуальный графический интерфейс (меню и диалоговые окна);
5. Компилировать и компоновать программы;
6. Производить отладку разрабатываемых программ и приложений в процессе их выполнения.

Visual C++ 2010 - одна из составных частей программы Visual Studio 2010 и является самодостаточной средой для разработки различных программ и приложений. Visual C++ 2010 включает следующие основные компоненты:

1. **Редактор** – для ввода, просмотра и проверки исходного кода;
2. **Компилятор** – для трансляции начального кода C++ в объектный код;
3. **Компоновщик** – создает выполняемые файлы, объединяя объектный код и библиотечные модули;
4. **Библиотеки** – сборники стандартных программ-модулей, которые можно применять в разрабатываемых программах и приложениях. Одна из самых важных библиотек - Microsoft Foundation Classes (базовые классы Microsoft или MFC), используемая при написании программ, работающих под управлением Microsoft Windows. Кроме этого, стандартные библиотеки C++ поддерживают операции ввода-вывода и другие стандартные возможности языка;
5. Прочие инструментальные средства, включая AppWizard (Мастер приложений), ClassWizard (Мастер классов) и Resource Editor (Редактор ресурсов).

2. Интерфейс программы Visual Studio 2010

Интерфейс Visual Studio 2010 нагляден и удобен для создания, редактирования, просмотра и отладки компонентов разрабатываемых программ и приложений – ресурсов, классов, файлов и др. Основной экран разделен на части с различными функциями, размеры которых можно изменять по своему усмотрению. Для облегчения решения стандартных задач есть контекстные меню, доступ к которым осуществляется с помощью нажатия правой кнопки мыши на различных компонентах изображения на экране.

С помощью Visual C++ 2010 можно работать с разрабатываемым приложением как с проектом.

Проект - это регламентированный (определенный протоколами Visual Studio 2010) набор файлов: заголовков, текстов программ, ресурсов, установок, конфигураций и др. Интерфейс Visual Studio 2010 дает возможность работать со всеми компонентами проекта одновременно, поэтому экран разделен на несколько зон (окон). Каждая разрабатываемая программа (приложение), даже самая простая, является проектом.

Набор папок и файлов, создаваемый средой Visual C++ 2010 при разработке проектов, называется **решением**.

При запуске программы Visual Studio 2010 открывается окно **Начальная страница – Microsoft Visual Studio (Администратор)**, состоящее из трех частей:

1. **Обозреватель решений** – окно в левой части экрана, в котором в дальнейшем будут представлены в виде дерева папки и файлы текущего проекта. В случае необходимости на месте данного окна можно открыть **Окно классов** данного проекта, его **Диспетчер свойств** или **Командный обозреватель**;
2. Окно **Начальная страница** в правой части экрана, в котором предложены на выбор возможные действия пользователя, в частности, **Создать проект** или **Открыть проект**, а также для открытия предложен список последних проектов;

3. Окно **Вывод** в нижней части экрана, в котором при построении решений проектов выводятся сообщения о предупреждениях (**warning**) и ошибках (**error**). На месте этого окна можно открыть **Окно определения кода** или окно **Результаты поиска**.

При создании консольных приложений на языке C++ работа осуществляется в окне **<Имя проекта> – Microsoft Visual Studio (Администратор)**, состоящем из трех частей:

1. **Обозреватель решений** (описание см. выше);
2. Окно для ввода программного кода и для работы с программой в правой части экрана;
3. Окно **Вывод** (описание см. выше).

3. Редактирование и отладка программ в Visual C++ 2010

Код (текст) программы вводится в окне редактора с использованием клавиатуры и основных приемов работы с текстом в операционной системе (ОС) Windows. При отображении текста программы используется синтаксическое раскрашивание. По умолчанию текст программы черного цвета с комментариями зеленого цвета и ключевыми и служебными словами синего цвета.

После того, как набран программный код, следует приступить к отладке программы. Для этого следует открыть меню **Построение** и выбрать пункт **Построить решение**. После чего все сообщения о предупреждениях и ошибках отображаются в окне **Вывод**. Ошибки следует исправлять обязательно, т. к. программа с ошибками выполняться не будет. С предупреждениями программа будет выполняться, однако их необходимо проанализировать и, если возможно, исправить или учесть. Описание ошибки находится в строке сообщения об ошибке (для получения подробного описания ошибки следует нажать **<F1>**). Чтобы исправить ошибку необходимо дважды щелкнуть в окне отладчика на сообщении о данной ошибке, после чего в окне редактора появится указатель на строке с ошибкой. После исправления всех ошибок необходимо открыть пункт меню **Отладка**. Программа запускается при выборе пункта меню **Начать отладку** или при нажатии клавиши **<F5>**.

4. Данные и переменные в языке C++

4.1. Типы данных в языке C++.

Основные или базовые типы данных в языке C++ следующие:

- int** – целочисленные данные (4 байта), диапазон значений: целые от -2 147 483 647 до 2 147 483 647;
- char** – символьные данные (1 байт), диапазон значений: от 0 до 255 или от -127 до 128;
- float** – числа с плавающей запятой (4 байта), диапазон значений: от 3.4E-38 до 3.4E+38;
- double** – число с плавающей запятой двойной точности (8 байт), диапазон значений: от 1.7E-308 до 1.7E+308;
- bool** – логические переменные (**true** и **false**).

Кроме вышеприведенных основных типов данных, в языке C++ используются еще несколько типов данных.

4.2. Объявление переменных и констант в языке C++

Фрагмент памяти, в котором хранится элемент данных и к которому можно обращаться по некоторому имени, называется **переменной**. Имена переменных могут включать буквы латинского алфавита **A – z** (в верхнем или нижнем регистре), цифры от 0 до 9 и знак подчеркивания. Имена переменных должны начинаться либо с буквы, либо со знака подчеркивания. В C++ принято назначать имена переменных с прописных букв, а классов – с заглавных. Компилятор C++ различает прописные и строчные буквы, например, **Sum** и **sum** означают разные переменные.

Объявление переменной с одновременным заданием типа хранимого под ее именем элемента данных осуществляется с использованием следующего синтаксиса:

ТипПеременной ИмяПеременной;

Например, строка **int arg;** объявляет целочисленную переменную с именем **arg**.

В языке C++ есть зарезервированные слова, имеющие специальное значение – **ключевые слова**. Это названия типов данных и некоторых операторов и др. Редактор среды разработки Visual C++ 2010 подсвечивает их **синим** цветом. Имена переменных не должны совпадать с ключевыми словами.

Объявляя переменную, можно сразу присвоить ей начальное значение. Например

```
int sum=0; или int sum(0);  
float a=2.7; или float a(2.7);
```

Переменная объявляется **перед** тем местом, где она будет впервые задействована. Подробнее о месте объявления переменных в C++ будет сказано ниже.

Переменную, не меняющую своего значения в программе, можно использовать как константу. В C++ объявление константы выглядит следующим образом:

```
const Тип ИмяКонстанты = Значение; .
```

Например, объявить постоянную **π** можно следующим образом:

```
const float pi = 3.14159; .
```

5. Выражения и операции в среде Visual C++ 2010

В языке C++ следующим уровнем представления данных после переменных и констант являются выражения.

Выражение – это некоторая допустимая комбинация переменных, констант, функций и знаков операций для вычислений в программах. Выражения в языке C++ записываются в строчку.

Например, формула

$$d = \frac{a + b(c + e)}{c(a + b) + t}$$

в языке C++ запишется следующим образом: $d = (a + b * (c + e)) / (c * (a + b) + t)$.

В приведенном выше выражении знак “=” обозначает операцию **присваивания**, которая выполняется следующим образом. Вычисляется значение выражения в правой части и присваивается переменной **d**.

В данной лабораторной работе будут использоваться привычные для студентов арифметические операции; в полном объеме операции среды Visual C++ 2010 приведены в лекционном материале по курсу и будут исследованы в следующих работах. Приоритет (очередность выполнения) арифметических операций такой же, как и в математике (см. таб. 1.1).

Математические действия с переменными, константами и функциями в языке C++ записываются только в строчку, при этом для соблюдения необходимой по условию задачи очередности операций используются круглые скобки.

Таблица 1.1. Арифметические

операции в языке C++

Операция	Знак в языке C++
Сложение	+
Вычитание	-
Умножение	*
Деление	/

6. Стандартные математические функции в среде Visual C++ 2010

В языке C++ в выражения можно вставлять стандартные математические функции, которые вызываются из библиотеки **<math.h>**. Перечень математических функций, которые чаще всего встречаются в вычислениях приведены в таблице 1.2.

Таблица 1.2. Основные стандартные математические функции (библиотека math.h)

Название функции	Что вычисляет	Тип данных функции и аргумента
abs(x)	Абсолютное значение (модуль) аргумента x	int abs(int x)
exp(x)	Экспонента e^x	double exp(double x)
log(x)	Натуральный логарифм $\ln x$	double log(double x)
log10(x)	Десятичный логарифм $\lg x$	double log10(double x)
pow(x,y)	Возведение в степень x^y	double pow(double x, double y)
sqrt(x)	Квадратный корень \sqrt{x} .	double sqrt(double x)
fmod(x,y)	Остаток от деления x/y	double fmod(double x, double y)
sin(x)	Синус (угол задается в радианах)	double sin(double x)
asin(x)	Арксинус (угол задается в радианах от -1 до +1)	double asin(double x)
cos(x)	Косинус (угол задается в радианах)	double cos(double x)
acos(x)	Арккосинус (угол задается в радианах от -1 до +1)	double acos(double x)
tan(x)	Тангенс (угол задается в радианах)	double tan(double x)
atan(x)	Арктангенс (угол задается в радианах)	double atan(double x)

При обращении к этим функциям необходимо придерживаться следующих правил:

- 1) **x** и **y** должны быть типа **double**;
- 2) углы (аргументы) в тригонометрических функциях задаются в радианах.
- 3) вычисляемые функциями данные имеют тип **double**.

Например, выражение $y = \sin^3(x^4) \frac{e^x + z^5 - 4.5 \cdot 10^2 \sqrt{x}}{\operatorname{tg}(a)(z^x + b)}$ на языке C++ будет иметь вид:

$y = \operatorname{pow}(\sin(\operatorname{pow}(x,4)),3) * (\exp(x) + \operatorname{pow}(z, 5) - 4.5 * 10 * 10 * \operatorname{sqrt}(x)) / (\tan(a) * (\operatorname{pow}(z,x) + b))$.


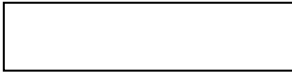
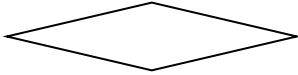




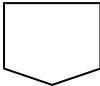

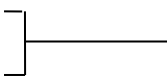
7. Алгоритмизация задач

Алгоритмизацию решаемых задач рассмотрим на примере программ с линейной структурой, которые являются самыми простыми и используются, как правило, для реализации вычислений по формулам. В программах с линейной структурой операторы выполняются последовательно один за другим. Алгоритмизация более сложных задач будет исследована при выполнении следующих лабораторных работ.

В схемах алгоритмов операционные блоки соединяются между собой линиями потока в виде стрелок. Допускается линии потока, которые идут сверху вниз и слева направо изображать без стрелок. При необходимости блок-схему могут сопровождать комментарии.

В таблице 1.3 представлены условные обозначения основных операционных блоков схем алгоритмов.

Таблица 1.3. Основные операционные блоки схем алгоритмов

№ п/п	Условное обозначение	Наименование	Описание операции
1		Начало, конец	Начало и конец алгоритма
2		Процесс	Вычислительная операция или их совокупность
3		Условие	Проверка условия, выбор дальнейшего направления процесса решения
4		Модификация	Заголовок цикла, проверка условий цикла
5		Ввод, вывод данных	Ввод исходных данных, вывод данных и результатов
6		Заранее определенный процесс	Использование ранее созданных алгоритмов, подпрограмм, функций
7		Печать документа	Вывод данных на печать
8		Соединитель внутрестраничный	Разрыв линий потока в пределах одной страницы
9		Соединитель межстраничный	Перенос линий потока на другую страницу
10		Узел	Слияние линий потока
11		Комментарии	Описание операционного блока и его особенностей

Исследуем процесс разработки алгоритма программы с линейной структурой на следующем примере.

Задача. Необходимо вычислить силу тока I в новогодней гирлянде, состоящей из $n=50$ электрических лампочек сопротивлением $r=20$ Ом. Используя закон Ома и формулу для расчета суммарного сопротивления последовательной цепи, составим блок-схему алгоритма (рис. 1.1).

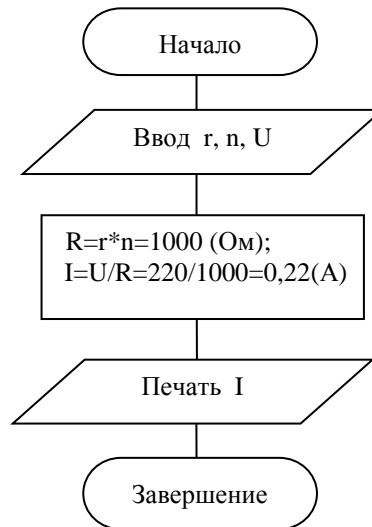


Рис. 1.1. Блок-схема алгоритма определения силы тока в новогодней электрогирлянде

8. Ввод и вывод данных в среде Visual C++ 2010

Лабораторные работы будут выполняться в Консольном приложении Win32 среды программирования Visual C++ 2010 консольный (с использованием клавиатуры и экрана дисплея) ввод данных производится при помощи оператора **cin**. В C++ этот оператор называется также потоком ввода. Например, для ввода значений трех переменных надо записать:

```
cin>>a >>b>>c;
```

где >> - символ операции извлечения данных из потока; **a**, **b** и **c** – переменные, значения которых будут вводиться. Вводимые значения должны разделяться пробелами, а ввод завершается нажатием клавиши <Enter>. Поточковый ввод и его операции автоматически распознают переменные и данные любого типа.

Консольный (на экран дисплея) вывод данных производится при помощи оператора **cout**. В C++ этот оператор называется также потоком вывода. Например, для вывода значений трех переменных надо записать:

```
cout<<a<<b<<c;
```

где << - символ операции вставки данных в поток; **a**, **b** и **c** – переменные, значения которых будут выводиться на экран. Поточковый вывод и его операции автоматически распознают переменные и данные любого типа. Вывод в Win32 производится в командную строку окна DOS. Помимо данных можно выводить и текстовую строку, заключив ее в кавычки:

```
cout<<"Summa a+b+c = "<<d;
```

Стандартные функции ввода и вывода находятся в библиотечном файле **iostream**. Чтобы связать программу разработчика со стандартной библиотекой ввода-вывода, необходимо в начале программы указать оператор подключения:

```
#include <iostream> .
```

Точка с запятой после операторов **include** не ставится.

Оператор **cout** часто используется с различными опциями (управляющими последовательностями) для расширения его функций.

ПРАКТИЧЕСКАЯ ЧАСТЬ

На диске d:\ создайте папку с названием и номером группы, в которой создайте папку с Вашей фамилией, а в ней – папку с номером лабораторной работы. Например, **d:\PE-11\Иванов\Лр1**. Выполнять эти и последующие действия с файлами и папками необходимо в файловом менеджере **Unreal Commander**.

Задание 1.1. Создайте проект **Lr1-1** для разработки программы, вычисляющей сумму трех целых чисел с использованием операторов консольных ввода и вывода данных. Проанализируйте код (текст) программы с комментариями, приведенный ниже.

Для этого:

1. Самостоятельно разработайте блок-схему алгоритма решения данного задания и аккуратно начертите в отчете (за основу возьмите схему на рис. 1.1).

2. Запустите Visual C++ 2010 (Пуск/Программы/Microsoft Visual Studio 2010 или ярлык Microsoft

Visual Studio 2010 на рабочем столе).

3. При создании нового проекта в Visual C++ 2010 необходимо выполнить команды основного меню **Файл**→**Создать**→**Проект**. В левой части появившегося диалогового окна выбрать установленный шаблон **Visual C++** →**Win32** и далее тип проекта – **Консольное приложение Win32**.

4. Ввести имя проекта **Lr1-1** в текстовое поле **Имя**, удалив перед этим надпись **<Введите имя>**.

5. Нажмите кнопку **Обзор**, выберите папку **d:\PE-11\Фамилия\Lp1** для размещения создаваемого проекта и нажмите **Ok**.

6. В открывшемся диалоговом окне **Мастер приложений Win32** выберите пункт **Параметры приложения** (слева) и тип приложения **Консольное приложение**. После этого нажмите на кнопку **Готово**.

7. Создать файл программного кода **Lr1-1.cpp**. Для этого в открывшемся окне **Lr1-1 - Microsoft Visual Studio** в центральной части с заголовком **Lr1-1.cpp** (окно редактора программного кода) ввести в уже созданную заготовку код (текст) программы (не нужно дублировать уже имеющиеся строки!).

```
#include "stdafx.h"
#include <conio.h> //Файл conio.h обеспечивает задержку окна DOS на
//экране дисплея
#include <iostream> //Директива include подключает файл заголовка
//iostream
using namespace std; //Подключает все имена из пространства имен std
int _tmain(int argc, _TCHAR* argv[ ]) //Объявление главной функции
{ //Начало главной функции
    double a; //Объявление переменной a
    double b, c=2, d=3; //Объявление переменных b, c, d и их
//инициализация
    cout<<"vvedite b"<<endl; //Вывод на экран надписи-приглашения vvedite b .
//Здесь endl – признак перемещения курсора на
//новую строку
    cin>>b; //Ввод значения переменной b
    a=b*(c+2*d-c*d)/(d*(5*c+4*b)); //Вычисление переменной a по заданной формуле
    cout<<"a = "<<a; //Вывод на экран значения a
getch(); //Функция задержки окна DOS на экране
return 0;
}
```

Внимание!! Студентам необходимо разобраться с действием и назначением каждой строки программного кода!!

8. После ввода программного кода в окно редактора тщательно проверьте введенный код на отсутствие ошибок, а затем откройте пункт меню **Построение** и щелкните левой клавишей мыши (ЛК) на команде **Построить решение**. **Visual C++ 2010** проведет анализ Вашей программы с выводом результатов этого анализа в окно **Вывод**. В последней строке выведенных результатов анализа будет указано, есть ли в программе ошибки.

9. Просмотрите текст в окне **Выводы**, где будут указаны характер ошибки и номер строки ее местонахождения. После устранения ошибок в программном коде снова выполните п. 7.

10. Если ошибки отсутствуют, откройте пункт меню **Отладка** и выполните команду **Начать отладку**.

11. В открывшемся окне DOS прочитайте инструкции, введите необходимые данные (подтверждая ввод данных нажатием клавиши **<Enter>**) и изучите полученные результаты. Результат выполнения **Задания 1** (проект **Lr1-1**) имеет следующий вид:

```
vvedite b
10
b+c+d = 15
```

12. Создайте в текстовом процессоре **Word** файл **Результат_Фамилия_Lp1**. Поля документа сделайте по 0,5 см.

13. Поместите окно DOS с результатами решения **Задания 1.1** в центральной части окна **Microsoft Visual Studio** ниже программного кода **Lr1-1.cpp** (см. рис. 1.2) и нажмите клавишу **<Prt Scr>**, после чего вставьте полученную копию экрана в файл **Результат_Фамилия_Lp1**. Над вставленным рисунком проставьте номер задания – **1-1**. Файл результатов не закрывайте до получения оценки за выполненную практическую часть работы в тетрадь с отчетом.

14. Закройте окно DOS.

15. Откройте пункт меню **Файл** и выполните команду **Закрывать решение**.

!Внимание! Следующие задания данной и других лабораторных работ выполняйте строго в соответствии с приведенным выше порядком действий!

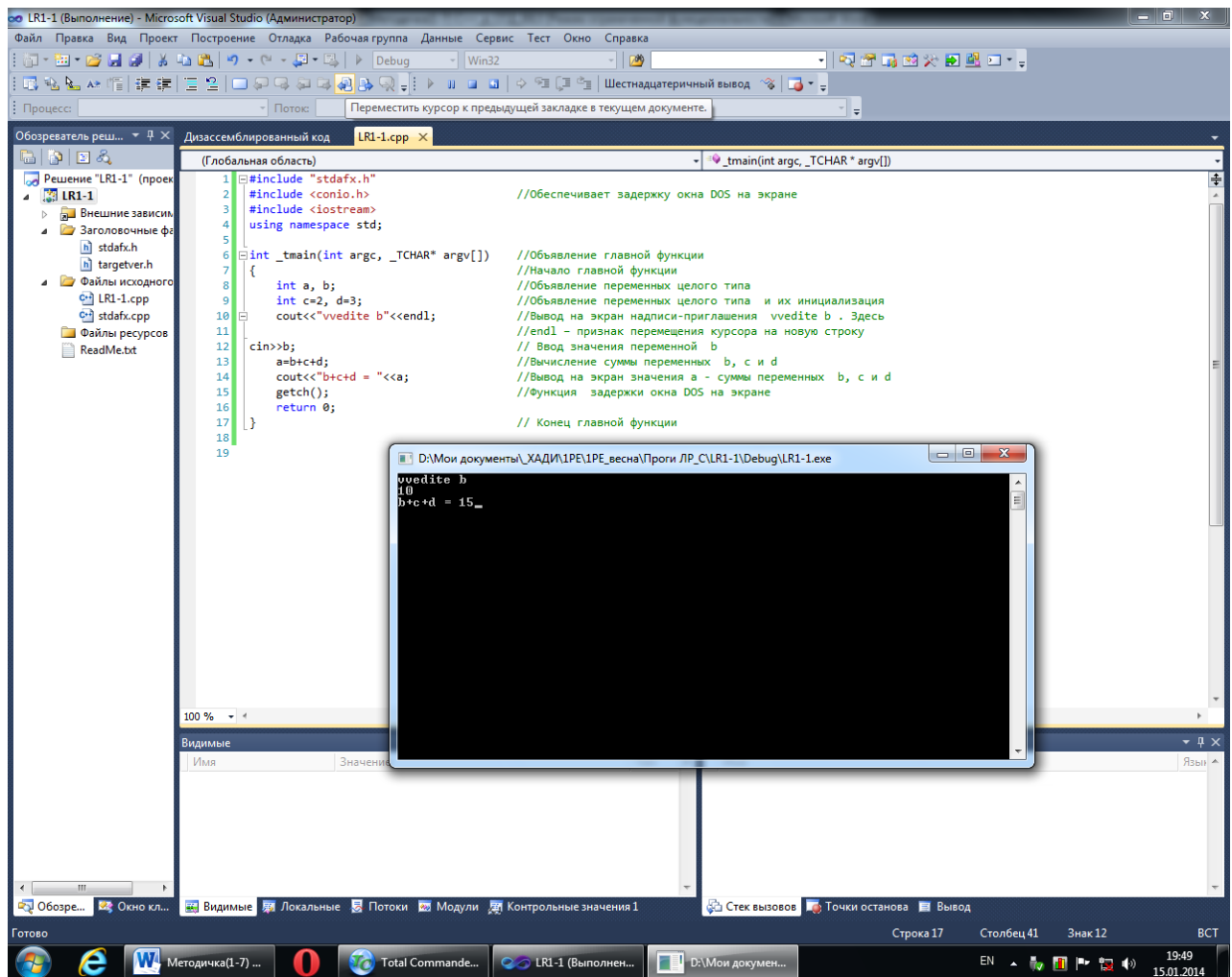


Рис. 1.2. Результаты выполнения Задания 1. Копия экрана с результатами выполнения проекта **Lr1-1**.

Задание 1.2. Создайте проект **Lr1-2** для вычисления переменной **a** по формуле $a = b \frac{c + 2d - cd}{d(5c + 4b)}$ с использованием операторов консольных ввода и вывода данных. Блок-схема алгоритма решения данного задания аналогична блок-схеме на рис. 1.1. Проанализируйте код (текст) программы с комментариями, приведенный ниже.

```

#include "stdafx.h"
#include <conio.h> //Файл conio.h обеспечивает задержку окна DOS
//на экране дисплея
#include <iostream> //Директива include подключает заголовочный
//файл iostream
using namespace std; //Подключает все имена из пространства имен std

int _tmain(int argc, _TCHAR* argv[]) //Объявление главной функции
{ //Начало главной функции
    double a; //Объявление переменной a
    double b, c=2, d=3; //Объявление переменных b, c, d и их
//инициализация
    cout<<"vvedite b"<<endl; //Вывод на экран надписи-приглашения vvedite b .
//Здесь endl – признак перемещения курсора на
//новую строку
    cin>>b; //Ввод значения переменной b
    a=b*(c+2*d-c*d)/(d*(5*c+4*b)); //Вычисление переменной a по заданной формуле
}
    
```



```

cout<<"a = "<<a;           //Вывод на экран значения a
getch();                   //Функция задержки окна DOS на экране
return 0;
}

```

Результат выполнения **Задания 2** (проект **Lr1-1**) имеет следующий вид:

```

vvedite b
10
a = 0.133333

```

Сохраните результаты выполнения **Задания 2** в виде копии экрана таким же образом, как и в **Задании 1**.

Задание 1.3. Самостоятельно создайте проект **Lr1-3** для вычисления площади поверхности и объема цилиндра по формулам

$$S = 2\pi r(h+r),$$

где **S** – площадь поверхности цилиндра; **r** – радиус основания цилиндра; **h** – высота цилиндра;

$$V = \pi r^2 h,$$

где **V** - объем цилиндра.

Радиус основания цилиндра равен номеру Вашего компьютера, а высота – удвоенному номеру компьютера.

Перед составлением программного кода **аккуратно (!)** начертите в отчете блок-схему алгоритма решения задачи, которая аналогична блок-схеме на рис. 1.1.

Примечание. Поскольку в C++ нет оператора возведения в степень, то для вычисления r^2 следует воспользоваться оператором умножения, менее трудоемким, чем функция **pow**.

Число π задайте как константу, а значения **r** и **h** введите с клавиатуры. Результат вычислений должен содержать необходимые пояснения. Например, для 10-го компьютера **V cilindra = 6 280**.

Здесь и далее сохраните результаты выполнения задания таким же образом, как и в **Задании 1**.

Задание 1.4. Создайте проект **Lr1-4** для вычисления переменной **Y** по формуле

$$Y = x^5 \frac{\sin^4 x^3 + 2e^{3a} - \operatorname{tg}\left(\frac{1-x}{1+x}\right)}{\cos 4(x+a)}$$

с использованием операторов консольных ввода и вывода данных. Код (текст) программы с комментариями приведен ниже. Переменные **x** и **a** зададим непосредственно в программном коде. Формулу для определения переменной **Y** запишем в виде, принятом в языке C++:

$$Y = \operatorname{pow}(x,5) * (\operatorname{pow}(\sin(\operatorname{pow}(x,3)),4) + 2 * \exp(3*a) - \tan((1-x)/(1+x))) / \cos(4*(x+a)) .$$

Блок-схема алгоритма решения данного задания аналогична блок-схеме на рис. 1.1. Код (текст) программы с комментариями приведен ниже.

// **Lr1-4.cpp**: определяет точку входа для консольного приложения.

```

#include "stdafx.h"
#include <conio.h>           // Файл conio.h обеспечивает задержку окна DOS
на экране дисплея
#include <iostream>         // Директива include подключает файл iostream
using namespace std;      // Подключает все имена из пространства имен std
int _tmain(int argc, _TCHAR* argv[]) // Объявление главной функции
{
    // Начало главной функции
    double Y;              // Объявление переменной Y
    double a=3, x=0.3;     // Объявление переменных a, x и их инициализация
    Y = pow(x,5) * (pow(sin(pow(x,3)),4) + 2 * exp(3*a) - tan((1-x)/(1+x))) / cos(4*(x+a)); // Вычисление Y
    cout << "Y = " << Y;    // Вывод на экран значения Y
    getch();               // Функция задержки окна DOS на экране
    return 0;
}

```

Результат вычислений:

$$Y = 48.8651 .$$

Задание 1.5. Создайте проект для вычисления функции Y по заданной формуле в соответствии со своим вариантом (номер компьютера). Варианты заданий находятся в таблице 1.4. Ввод исходных данных организуйте непосредственно в программе (см. задание 4).

Перед разработкой программного кода запишите заданную формулу в отчете на языке C++.

Таблица 1.4 Исходные данные и формулы для расчета Y (Задание 1.5)

№ варианта	Формула для расчета Y	Значения x, a, b
1	$Y = \operatorname{tg} \frac{1-x}{1+x} + \sin^2 5x + e^{5a}$	$x=0,5; \quad a=3,5$
2	$Y = \frac{\sin \frac{x+1}{4}}{\sin^2 5x + e^{3a}}$	$x=0,7; \quad a=1,5$
3	$Y = \frac{\sin^3(x+a) - \cos^2(x+a)}{(x+a)^4}$	$x=0,82; \quad a=2,55$
4	$Y = \frac{\operatorname{tg}^3(x+a) - \arccos^2(x+a)}{(x+a)^4}$	$x=0,68; \quad a=5,55$
5	$Y = \operatorname{ctg} \frac{1-3x}{1+2x} + \cos^2 5x + e^{7a}$	$x=0,35; \quad a=4,8$
6	$Y = \frac{\cos^3(x+a) - 7(x+a)}{\operatorname{tg}(x+a)^4}$	$x=0,62; \quad a=4,55$
7	$Y = \frac{\cos \frac{3a+1}{4}}{\sin^3 3x + e^{4a}}$	$x=0,43; \quad a=2,6$
8	$Y = \frac{\sin^3(x+a) - \cos^2(x+a)}{(x+a)^4}$	$x=0,74; \quad a=1,55$
9	$Y = \frac{\operatorname{tg} \frac{4a^2+1}{4}}{\cos^3 2x + e^{2a}}$	$x=0,14; \quad a=2,55$
10	$Y = \sin \frac{1-x}{1+x} + \operatorname{tg}^4 5x + e^{5a}$	$x=0,34; \quad a=4,95$
11	$Y = \frac{\sin^3(x+a) - \arccos^2(x+a)}{\cos(x+a)^4}$	$x=0,14; \quad a=2,95$
12	$Y = \frac{\operatorname{ctg} \frac{x^3+1}{4}}{\cos^2 5x + e^{3a}}$	$x=0,75; \quad a=1,9$
13	$Y = \frac{\operatorname{ctg}^3(3x+a) - \sin^2(x+7a)}{(5x+a)^3}$	$x=0,44; \quad a=2,95$
14	$Y = \frac{\arcsin^3 \frac{4x+1}{4}}{\operatorname{ctg}^2 3x + e^{3a}}$	$x=0,27; \quad a=1,9$

15	$Y = \frac{\arcsin^3 \frac{4x+1}{4}}{\operatorname{ctg}^2 3x + e^{3a}}$	x=0,49; a=3,7
16	$Y = \frac{\sin^3(x+a) - \cos^2(x+a)}{(x+a)^4}$	x=0,83; a=4,7
17	$Y = \frac{\operatorname{arcctg} \frac{2x^3+1}{4}}{\cos^2 5x + e^{3a}}$	x=0,37; a=2,75
18	$Y = \frac{\operatorname{tg}^3(x+a) - 5(\sin x + a)}{\sin^3(x+a)^4}$	x=0,13; a=0,7

Задание 1.6. Создайте проект для вычисления функции Y по заданной формуле в соответствии со своим вариантом (номер компьютера). Варианты заданий находятся в таблице 1.5. Ввод исходных данных организуйте непосредственно в программе (см. задание 4). Перед разработкой программного кода запишите заданную формулу в отчете на языке C++.

Таблица 1.5 Исходные данные и формулы для расчета Y (Задание 1.6)

№ варианта	Формула для расчета Y	Значения x, a, b
1	$Y = \operatorname{tg} \frac{1-x}{1+x} + \sin^2 5x + e^{5a}$	x=0,5; a=3,5
2	$Y = \frac{1 + \cos \frac{1-x^2}{1+x^2}}{\sqrt{1 + \left(\frac{1-a^3}{1+a^4}\right)^2}} - e^{-\frac{x^2}{2}}$	x=1,573; a=1,775
3	$Y = \ln\left(1 + \sqrt{1+x^2}\right) + \frac{1+x^2}{\sqrt{1 + \frac{a^4}{2}}}$	x=1,573; a=1,775
4	$Y = \frac{\sin^3(a+x^2)}{\sqrt{a + \frac{x^3+1}{2} + \left(\frac{x^2+1}{2}\right)^5}}$	x=1,573; a=1,775
5	$Y = \left(a + \frac{a}{a^4 + x^2}\right) - \sqrt{1 + \frac{a}{a^2 + x^2}}$	x=1,573; a=1,775
6	$Y = 2a^3 + \frac{2x^4}{\sqrt{1+a^2}} - e^{-\frac{a^2+1}{2}}$	x=1,573; a=1,775
7	$Y = \frac{e^{\frac{x^2}{2}} + e^{\frac{1+a^2}{2}}}{1 + \frac{a^2}{2} + \left(\frac{x^4}{2}\right)^2}$	x=1,573; a=1,775
8	$Y = \lg\left(1 + \sqrt{1+a^5}\right) + \frac{e^a + x^4}{\sqrt[4]{1 + \frac{x^3}{2}}}$	x=1,573; a=1,775

9	$Y = \frac{(a+x)^3 \cdot \ln \frac{a+x}{2}}{\sqrt[3]{1 + \frac{(a+x^3)^2}{4}}}$	x=1,573; a=1,775
10	$Y = 2x^3 + \frac{2a^4}{\sqrt[3]{1+x^3}} - e^{-\frac{a^2+1}{2}}$	x=1,573; a=1,775
11	$Y = 4 \sqrt{\frac{1 + \operatorname{tg}^2 \frac{x+1}{4}}{1 + \frac{a^3+1}{4}}} - e^{-\frac{x+1}{4}}$	x=1,573; a=1,775
12	$Y = \frac{e^{\frac{5x^2}{2}} + e^{\frac{1+4a^3}{2}}}{1 + \frac{x^2}{2} + \left(\frac{a^3}{2}\right)^2}$	x=1,573; a=1,775
13	$Y = \frac{\lg(1+x^4)}{\sqrt{1 + \frac{x^2+a^3}{2} + \left(\frac{x^2+1}{2}\right)^2}}$	x=1,573; a=1,775
14	$Y = \frac{1 + \left(\frac{x}{a}\right)^{2,5} + \sqrt{1 + \left(\frac{x}{a}\right)^{2,5}}}{1 + e^{x^3} + \sqrt{1 + \left(\frac{x}{a}\right)^{2,5}}}$	x=1,573; a=1,775
15	$Y = \left(x^3 + \frac{a}{a^2+x^2}\right) - \sqrt{1 + \frac{a}{a^2+x^4}}$	x=1,573; a=1,775
16	$Y = 4 \sqrt{\frac{1 + \operatorname{tg}^2 \frac{a^3+1}{4}}{1 + \frac{x+1}{4}}} - e^{-\frac{a^2+1}{4}}$	x=1,573; a=1,775
17	$Y = \frac{1 + \left(\frac{x^3}{a}\right)^{2,5} + \sqrt{1 + \left(\frac{x}{a}\right)^{2,5}}}{1 + e^x + \sqrt{1 + \left(\frac{x}{a^3}\right)^{2,5}}}$	x=1,573; a=1,775
18	$Y = \ln\left(1 + \sqrt{1+x^2}\right) + \frac{1+x^2}{\sqrt[3]{1 + \frac{a^3}{2}}}$	x=1,573; a=1,775

Контрольные вопросы

1. Что такое проект в программе Visual Studio 2010?
2. Перечислите основные или базовые типы данных в языке C++.
3. Раскройте понятие переменной в языке C++.
4. Раскройте понятие выражения в языке C++.
5. Как записываются математические действия с переменными, константами и функциями в языке C++?
6. Каковы правила обращений к математическим функциям в языке C++?
7. Чем отличаются программы с линейной структурой?
8. Как вывести на экран значение переменной?

9. Как в одном операторе объявить тип переменной и задать ее значение?
10. Перечислите опции оператора **cout** и раскройте их действие.
11. Какие функции выполняет оператор **include**?
12. Какие функции выполняет файл заголовка **iostream**?