МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Информатика»

Лабораторная работа №1

**«Основные средства и технология разработки**

**консольных программных проектов**

**в интегрированной среде Visual Studio .NET»**

**по дисциплине**

**«Алгоритмизация и программирование»**

Выполнил: студент гр. БИБ2305 Пуховский С.Д.

Вариант №1

Проверил: доц. Юсков И.О.

Москва, 2023 г

**Задание**

1) Изучить основные средства и технологию разработки консольных программных проектов в интегрированной среде Visual Studio .NET.

2) Выполнить все шаги по созданию первого консольного проекта на VС++ из п. 2.3 «Пример выполнения задания для изучения технологии работы в среде Microsoft Visual Studio .NET 2022»

3) Оформить отчет по работе и представить его преподавателю.

4) Ответить на контрольные вопросы, замечания преподавателя по работе и на заданные им вопросы по теме.

5) Получить отметку о выполнении и защите работы

**Ход выполнение работы**

В окне Создать проект из списка установленных шаблонов проектов необходимо выбрать Visual C++, а затем указать тип проекта, выделив шаблон Консольное приложение Win32. В этом же окне надо указать имя и расположение вновь создаваемого проекта

В поле Имя укажите имя проекта, например, Project или любое другое допустимое имя.

В поле Расположение необходимо указать путь к папке, в которую вы хотите поместить ваш проект. Этот путь можно прописать вручную вновь выпавшем окне Расположение проекта выбрать созданную ранее папку (для работы в лабораторном классе это должна быть папка Общие документы), нажать кнопку Выбор папки, и вернуться в окно Создать проект. 2.6) Проверьте, у вас должно быть установлено:

• в окне Шаблоны – шаблон Консольное приложение Win64;

• в текстовом поле Имя указано имя создаваемого вами проекта;

• в текстовом поле Расположение указан путь к папке, в которой вы хотите создать ваш проект;

• переключатель Создать каталог для решения выключен (сброшена галочка);

• поле Имя решения недоступно

Если все в порядке, нажмите кнопку OK. 2.8) На экране появится окно Мастер приложений Win64 –Project1. В результате чего у нас получилась готовая форма для выполнения последующих действия (см. рисунок 1)

**ГЛАВА 1.1**

«Создание проекта»

Выбираем «Консольное приложение» из предложенных шаблонов (см. рис. 1.1).

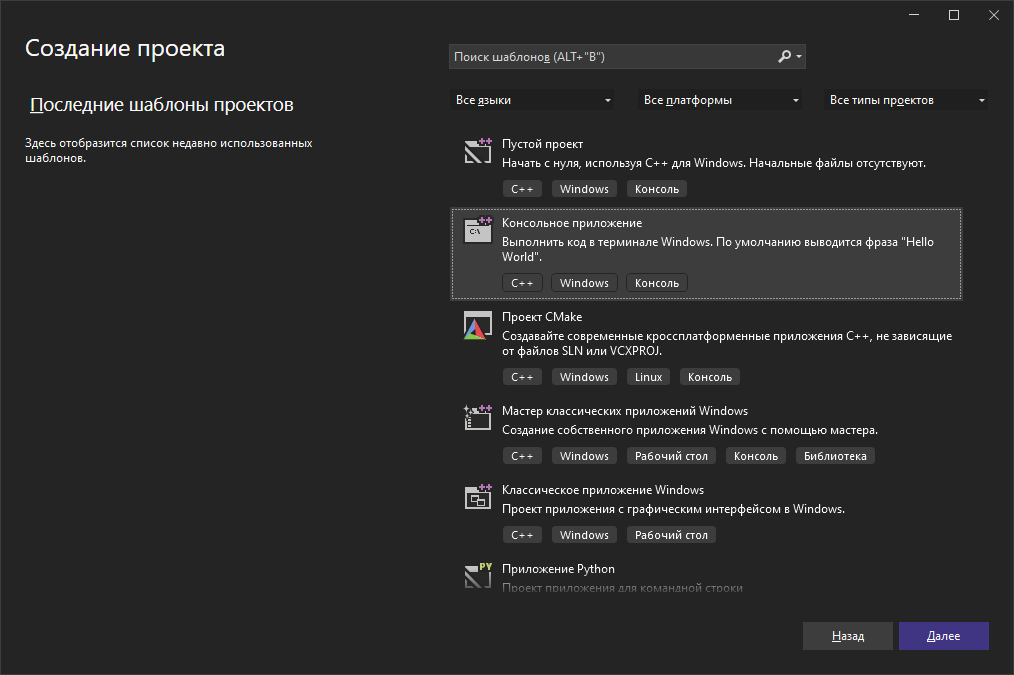


Рисунок 1.1 – Выбор шаблона

Пишем код необходимой программы (см. рис. 1.2).

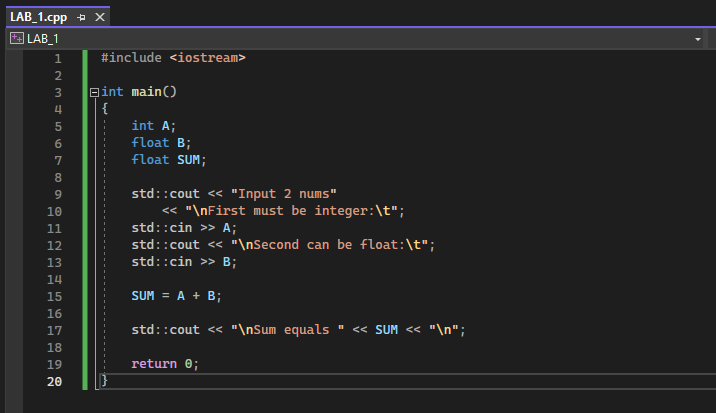


Рисунок 1.2 – Код программы

Выполняем компиляцию программного кода (см. рис. 1.).

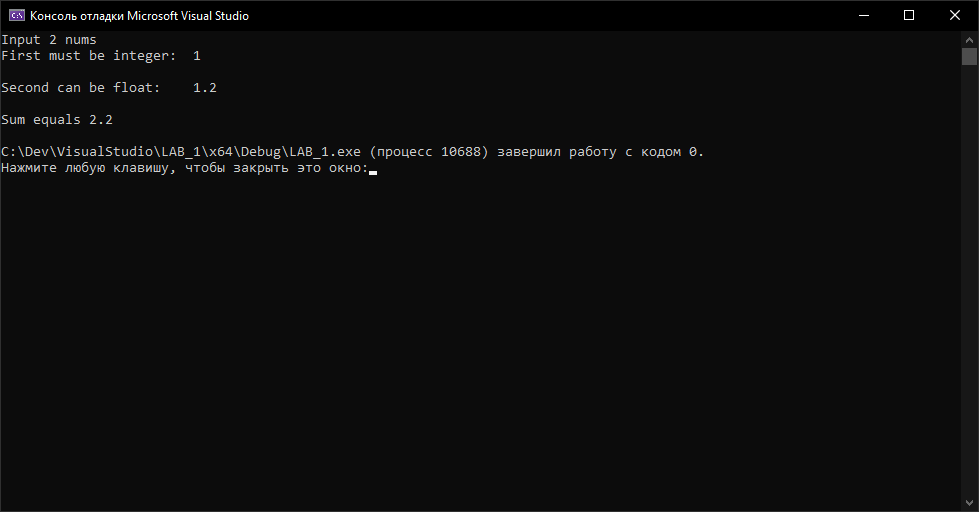


Рисунок 1.3 – Пример успешной компиляции и запуска программы

**ГЛАВА 1.2**

«Работа алгоритма»

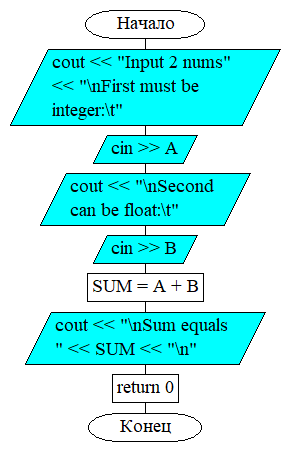
Данный алгоритм (см. рис. 2.1) принимает на вход два значения: целочисленное значение и значение с плавающей точкой. После ввода значений алгоритм складывает их в число с плавающей точкой.

Рисунок 2.1 – алгоритм программы

В результате выполнения программы пользователь получает ответ (см. рис. 2.2) без потери дробной части второго введенного значения.

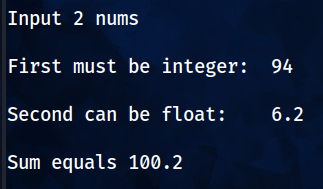


Рисунок 2.2 – пример штатного выполнения программы

**ГЛАВА 1.3**

«Тестирование программы»

Так как программа на вход принимает одно целочисленное значение (первое) и одно дробное значение в виде числа с плавающей точкой, то мы можем попробовать ввести значения, не предусмотренные программой что бы проверить как она будет работать. Стоит заметить, что второе вводимое число может быть целочисленным, так как его можно представить в виде дробного числа, дробная часть которого равна нулю.

Рассмотрим следующие ситуации:

1. Первое введенное значение дробное
2. Первое введенное значение не является числом
3. Второе введенное значение не является числом

Проверка ситуации №1:

Введем в качестве первого значения число с плавающей точкой.

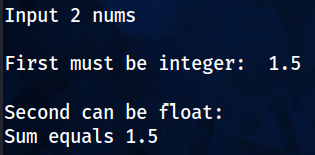


Рисунок 3.1 – результат выполнения программы

после проверки первой ситуации

Программа завершилась без критических ошибок, однако у пользователя не запросили второе значение для ввода (см. рис. 3.1).

Объяснение: После ввода в качестве целочисленного значения дробного числа программа присвоила целую часть переменной первого вводимого числа и оставила дробную часть в потоке пользовательского ввода, после чего она попала в пользовательский ввод второго вводимого числа и, так как, этот ввод принимает дробные значения то оставшаяся дробная часть сразу была присвоена переменной второго вводимого числа.

В качестве ответа мы получили введенное дробное значение.

Проверка ситуации №2:

Введем в качестве первого значения символ, не являющийся числом.

Для примера это будет буква «Т».

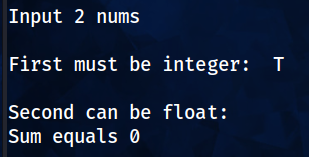


Рисунок 3.2 – результат выполнения программы

после проверки второй ситуации.

Программа завершилась без критических ошибок, однако в качестве ответа мы получили ноль (см. рис. 3.2).

Объяснение: В отличии от первой ситуации, где дробная часть из потока пользовательского ввода попала в ввод, допускающий значения с плавающей точкой, в нашей программе нет ввода, допускающего значения, не являющегося числом. После попытки принять на вход наше значение каждый ввод ничего не присваивал переменным, и они оставались равными нулю.

В качестве ответа мы получили сумму нулей, то есть ноль.

Проверка ситуации №3:

Введем в качестве первого значения символ, не являющийся числом.

Для примера это будет буква «К».

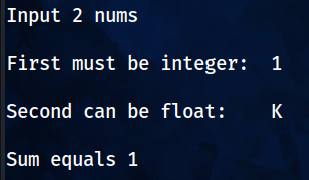


Рисунок 3.3 – результат выполнения программы

после проверки третьей ситуации.

Программа завершилась без критических ошибок, однако в качестве ответа мы получили значение первого вводимого числа (см. рис. 3.3).

Объяснение: Первое число полностью подходит под условия запрашиваемого числа и присваивается переменной первого вводимого числа, после чего со вторым вводом повторяется ситуация из примера №2 (см. рис. 3.2).

В качестве ответа мы получили только первое вводимое число.

**ГЛАВА 1.4**

«Процесс отладки программы»

Так как программа работает правильно добавим несколько ошибок вручную и проверим как будет себя вести компилятор.

Ошибки будут следующие:

1. Объявление неиспользуемой переменной
2. Замена операции присваивания операцией сравнения
3. Исключение из кода объявления переменной «В»
4. Исключение из кода подключения библиотеки «iostream»
5. Замена >> на <<
6. Перемещение объявления переменной после ее использования

Ошибка №1

Объявление неиспользуемой переменной (см. рис. 4.1).

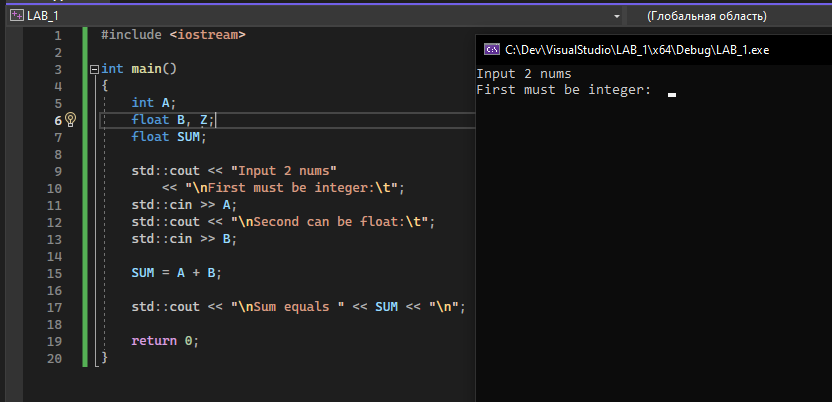


Рисунок 4.1

Программа продолжает компилироваться без критических ошибок (выдается только warning, который можно игнорировать) и нормально запускается.

Объявление неиспользуемой переменной не является ошибкой так как это всего лишь нерациональное использование ресурсов памяти компьютера.

Ошибка №2

Замена операции присваивания операцией сравнения (см. рис. 4.2).

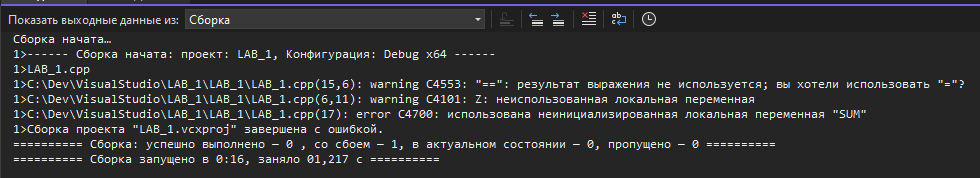


Рисунок 4.2

Программа не компилируется. Компилятор подсказывает что скорее всего нужно использовать не «=» а «==», так как в противном случае программа не будет работать.

Ошибка №3

Исключение из кода объявления переменной «В» (см. рис. 4.3).

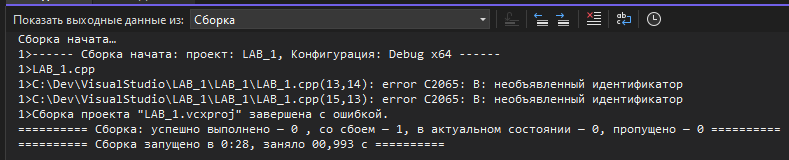


Рисунок 4.3

Программа не компилируется. Без объявленной переменной «В» мы не можем в нее что-либо записывать и использовать так как переменной нет в памяти.

Ошибка №4

Исключение из кода подключения библиотеки «iostream» (см. рис. 4.4).

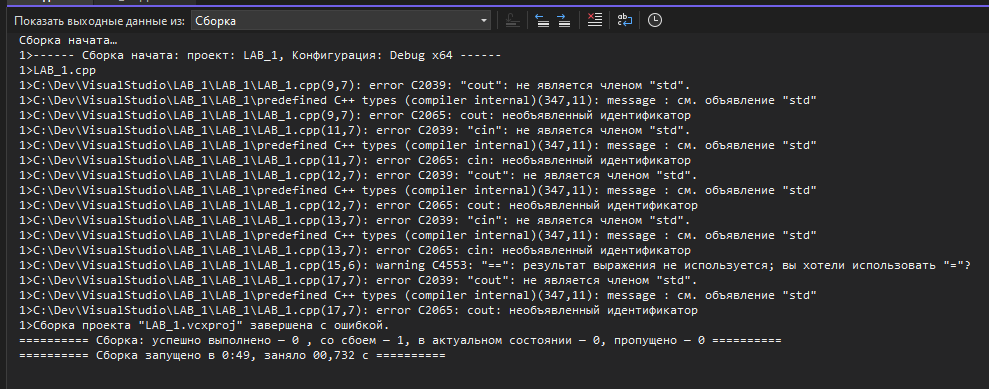


Рисунок 4.4

Программа не компилируется. Отключенная библиотека отвечает за функции «cout» и «cin», которые мы используем. Без них программа не может работать так как не понимает, что мы хотим сделать, ведь без библиотеки функционал функции не описан.

Ошибка №5

Замена >> на << (см. рис. 4.5).

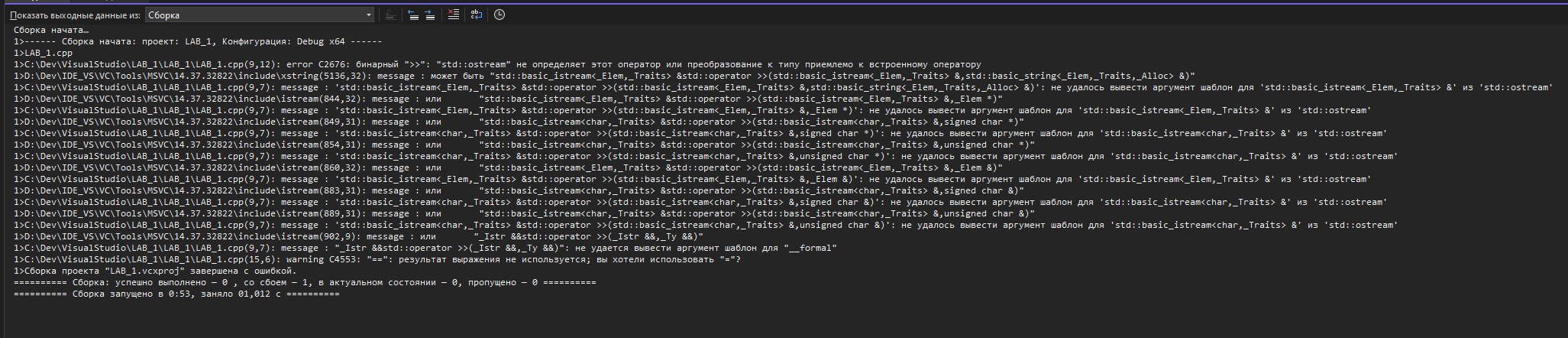


Рисунок 4.5

Программа не компилируется. Операнд «>>» отвечает за присвоение полученного из ввода значения переменной, а «<<» за вывод переданной в функцию информации в консоль. Если мы меняем их местами, то мы либо не можем присвоить значение так как его нечему присваивать, либо не можем вывести информацию так как нечего выводить.

Ошибка №6

Перемещение объявления переменной после ее использования (см. рис. 4.6).

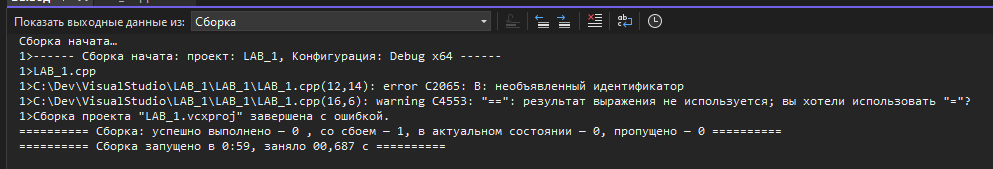


Рисунок 4.6

Программа не компилируется. Мы пытаемся присвоить значение необъявленной переменной, которой нет в памяти устройства, что невозможно.