

### Практическое занятие № 3

**Тема практического занятия:** Основные операторы языка программирования.

**Цель практического занятия:** овладение навыками использования и применения основных операторов языка программирования для решения задач и разработки программ.

**В результате выполнения данной работы обучающийся должен уметь:**

1. Использовать оператор присваивания.
2. Уметь использовать условные операторы, в том числе оператор множественного выбора.
3. Уметь использовать оператор цикла.
4. Обеспечивать взаимодействие с пользователем, позволяя ему вводить данные и просматривать результаты.

**Знать:**

1. Синтаксис оператора присваивания.
2. Синтаксис условных операторов.
3. Синтаксис операторов цикла.
4. Операторы ввода-вывода для взаимодействия с пользователем.
5. Операторы преобразования типов данных.
6. Как правильно использовать эти операторы для решения задач, разрабатывать алгоритмы и написание программ с использованием базовых операторов языка программирования

**Перечень оборудования, необходимого для выполнения задания:**

- Автоматизированные рабочие места по количеству обучающихся (процессор Intel Core i7 или аналогичный, БП 700 Вт, 32 Гб ОЗУ, SSD 512 Гб, HDD 4 ТБ SATA 7200 rpm, RTX 3060 12GB);
  - Автоматизированное рабочее место преподавателя (процессор Intel Core i7 или аналогичный, БП 700 Вт, 32 Гб ОЗУ, SSD 512 Гб, HDD 4 ТБ SATA 7200 rpm, RTX 3060 12GB);
  - Монитор 34", изогнутый, 3440x1440, 6 ms, 178°/178°, 300 cd/m2, 20M:1, HDMI, DP, USB, регулировка по высоте (1 монитор на одно рабочее место).
  - Интерактивная панель 86" с OPS ПК
- Программное обеспечение:
- Microsoft Visual Studio.

**Общие теоретические сведения:**

Оператор присваивания - это оператор в языке программирования, который используется для присваивания значения переменной или

выражения. Он позволяет изменить значение переменной на новое значение или связать переменную с результатом вычисления выражения.

Оператор присваивания позволяет программисту обновлять значения переменных в процессе выполнения программы, а также использовать предыдущие значения переменных для вычисления новых значений.

Условный оператор - это конструкция в языке программирования, которая позволяет выполнять различные действия в зависимости от условия или значения логического выражения. Он позволяет программе принимать решения во время выполнения на основе некоторого условия.

Оператор множественного выбора - это конструкция в программировании, которая позволяет выполнить различные действия в зависимости от значения выражения или переменной. Оператор множественного выбора обычно используется для замены множества вложенных условных операторов.

Оператор цикла - это конструкция в программировании, которая позволяет многократно выполнять один и тот же блок кода до тех пор, пока определенное условие остается истинным, либо пока не будет достигнута определенная оговоренная точка выхода.

Операторы цикла широко используются для автоматизации повторяющихся задач и обработки коллекций данных.

Существуют разные типы операторов цикла в разных языках программирования, но наиболее популярные из них: цикл с предусловием, цикл с постусловием, цикл с параметром и цикл перебора.

Операторы ввода-вывода — это инструкции или функции в языках программирования, которые позволяют программе взаимодействовать с внешним миром, вводить данные из внешних источников и выводить результаты работы программы на экран или другие устройства. Эти операторы позволяют программе обмениваться информацией с пользователем или другими программами. Они часто используются для чтения и записи файлов, взаимодействия с сетевыми службами, обработки пользовательского ввода и отображения результатов.

Операторы преобразования типов (или приведения типов) - это инструкции в языках программирования, которые позволяют изменить тип значения переменной на другой тип данных. Преобразование типов может быть необходимо, когда требуется привести значение одного типа к другому для выполнения определенных операций или соблюдения правил языка программирования.

### **Задание:**

1. Разработайте программу, которая будет играть роль калькулятора. Пользователь должен иметь возможность ввести два числа, операцию между ними и получить результат выполнения операции.

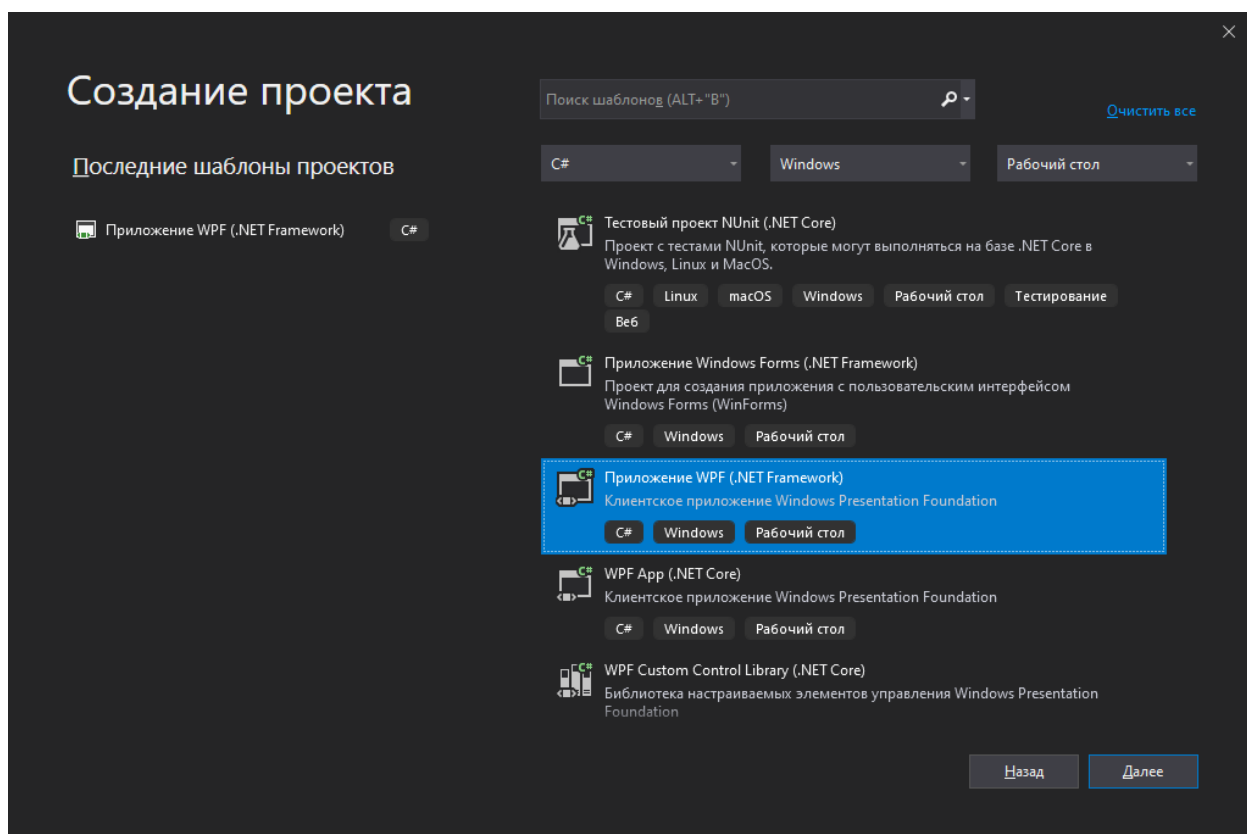
2. Предусмотрите выполнение следующих операций: сложение, вычитание, умножение, деление, возведение в степень, факториал. Не допускается делить на 0, в этом случае следует в качестве результата выдать ошибку.
3. Факториал должен вычисляться только для одного числа: того, которое указано первым. Число должно быть целым.
4. Если пользователь вместо операции введет что-то другое, необходимо в качестве результата выдать ошибку.

### Указания по технике безопасности:

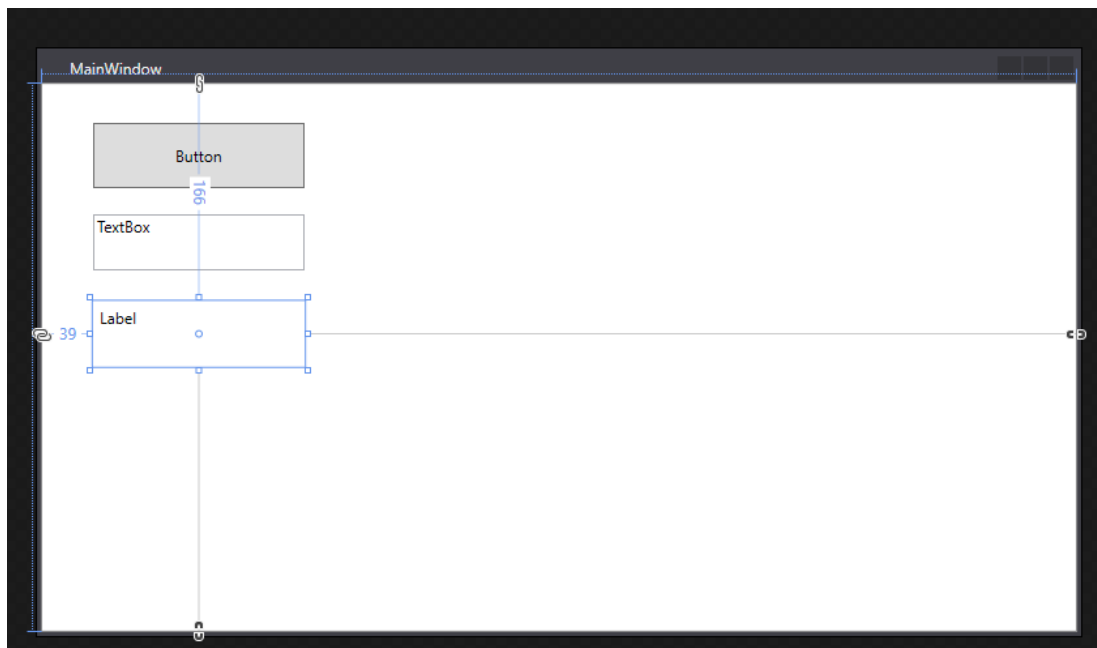
Инструкция по технике безопасности при работе в лаборатории, оборудованной компьютерной техникой.

**Технология выполнения работы (этапы, последовательность действий):**

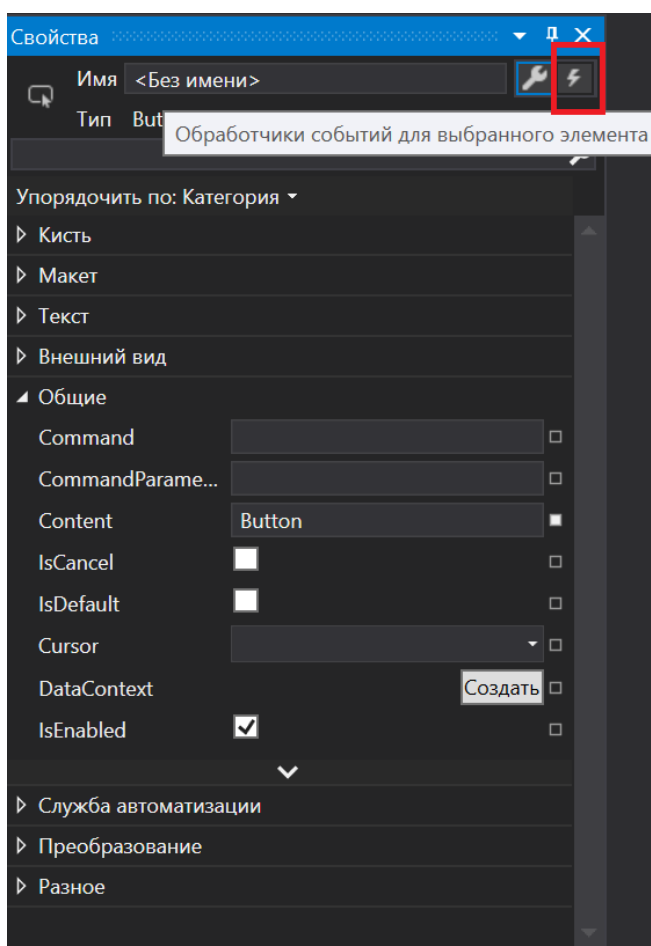
1. Запустите среду разработки и создайте проект.



2. В главном окне на одной линии разместите три текстовых поля: левый операнд, операцию, правый операнд. Справа от них разместите кнопку с надписью «=» и метку или текстовое поле для выдачи результата.  
На снимке экрана показано, как выглядят кнопка, текстовое поле и метка:



3. Создайте обработчик события нажатия кнопки.



4. Реализуйте в обработчике калькулятор. При проверке операции используйте оператор множественного выбора. Пример оператора множественного выбора:

```

/// <summary>
/// Метод выполняет вычисления в зависимости от выбранного действия
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
Ссылка: 0
private void Button_Click(object sender, RoutedEventArgs e)
{
    int length = Convert.ToInt32(tbLength.Text);    //определить длину комнаты
    int width = Convert.ToInt32(tbWidth.Text);      //определить ширину комнаты
    MyRoom.SetLength(length);
    MyRoom.SetWidth(width);

    //выбрать действие
    switch(cbOption.SelectedIndex)
    {
        //периметр
        case 0:
            lResult.Content = "Периметр комнаты: " + MyRoom.RoomPerimeter();
            break;
        //площадь
        case 1:
            lResult.Content = "Площадь комнаты: " + MyRoom.RoomSquare();
            break;
        //тип
        case 2:
            lResult.Content = "Перед вами - " + MyRoom.TypeRoom(MyRoom.TypeRoomCode());
            break;
    }
}

```

5. Для вычисления факториала используйте цикл. Пример оператора цикла:

```

private void Calculate_Click(object sender, RoutedEventArgs e)
{
    myRoom.SetLength(Convert.ToInt32(tbLength.Text));
    myRoom.SetWidth(Convert.ToInt32(tbWidth.Text));
    int n = Convert.ToInt32(tbTime.Text);
    lResult.Content = "";
    int F, F1, F2;
    F = F1 = F2 = 1;
    for(int i=3;i<=n;i++)
    {
        F = F1 + F2;    //вычисление числа Фибоначчи
        F2 = F1;        //запоминание нового числа, предшествующего предыдущему
        F1 = F;         //запоминание нового предыдущего числа
    }
    lResult.Content = "Всего " + F + " пар.";
}

```

6. Проверьте работоспособность вашего приложения, вводя различные данные, в том числе и некорректную операцию. Убедитесь в том, что результаты вычислений всех действий совпадают с ожидаемыми.

### Требование к отчету:

1. Приложение собирается и компилируется без ошибок, запускается.
2. Сложение двух чисел выполняется корректно.
3. Вычитание двух чисел выполняется корректно.
4. Умножение двух чисел выполняется корректно.

5. Деление двух ненулевых чисел выполняется корректно.
6. При введении в качестве второго операнда деления 0 выдается сообщение об ошибке.
7. Возведение в степень выполняется корректно.
8. Вычисление факториала выполняется корректно.
9. В случае ввода недопустимой операции выдается сообщение об ошибке.

#### **Контрольные вопросы:**

1. Что такое оператор присваивания? Укажите, где в вашем приложении операторы присваивания. Какие арифметические операции вы использовали?
2. Расскажите о совместимости типов данных. Что такое преобразование типов, как оно используется?
3. Как в вашем приложении обеспечивается совместимость типов?
4. Как в вашем приложении обеспечивается взаимодействие с пользователем?
5. Что такое условный оператор?
6. Что такое оператор множественного выбора? Чем он отличается от условного оператора, когда его целесообразно использовать?
7. Что такое условие? Как формируется условие в условном операторе, какие возможны операции?
8. Покажите, где в вашем приложении используется условный оператор?
9. Что такое оператор цикла, какие виды циклов вы знаете?
10. Для чего использовался цикл в вашем приложении? Какой тип цикла был использован и почему?

#### **Основные и дополнительные источники, электронные ресурсы:**

1. Подбельский, В. В. Язык C#. Базовый курс: учебное пособие / В. В. Подбельский. - 2-е изд., перераб. и доп. - Москва: Финансы и статистика, 2022. - 408 с. - ISBN 978-5-00184-079-4. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1913989>
2. Гуриков, С. Р. Основы алгоритмизации и программирования на Python: учебное пособие / С.Р. Гуриков. — Москва: ИНФРА-М, 2023. — 343 с. — (Среднее профессиональное образование). - ISBN 978-5-16-016906-4. - Текст: электронный. - URL: <https://znanium.com/catalog/product/1927269>.