МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

ОТЧЕТ ПО вычислительной (ознакомительной)

практике

Кукулы Дмитрия Игоревича

студента 1 курса 8 группы,

специальность «Прикладная математика»

Руководитель практики:

старший преподаватель

Колб О.О.

Минск, 2023

ОГЛАВЛЕНИЕ

[Введение 3](#_Toc139735800)

[Глава 1 Теоретические основы ПРОГРАммирования по направлению .NEt C# development 4](#_Toc139735801)

[1.1 Общая характеристика курса 4](#_Toc139735802)

[1.2 Модули курса 4](#_Toc139735803)

[Глава 2 Практическая часть курса 7](#_Toc139735804)

[Заключение 10](#_Toc139735805)

[Список использованных источников 11](#_Toc139735806)

Введение

В период с 26 июня по 8 июля было предложено пройти учебную ознакомительную практику для студентов первого курса. За это время студенту необходимо было выполнить ознакомительную, практическую и проектную части.

В ознакомительной части было описаны базовые знания языка программирования С#, основные методы, структуры, типы данных и алгоритмы.

Практическая часть включает в себя курс, состоящий из 18 модулей, каждый из которых затрагивает тот или иной раздел языка, начиная от синтаксиса и заканчивая потоками.

Проектная часть – собственная работа студента по интересующей его теме. Проект ставит задачи актуальности, неповторимости, технологической сложности и другие.

Все части ставят основные цели и задачи практики, которые достигаются по мере прохождения.

* Цель практики: изучить и ознакомиться с языком программирования С#, научиться взаимодействовать с инструментом Git. Углубить знания в теории программирования.
* Задачи:

1. Изучить инструменты для совместной разработки проектов и контроля версий Git.
2. Прохождение курса по C#

Для прохождения практики было необходимо авторизоваться в ряде тестирующих систем:

* Training.by: [dima.kukula.04@mail.ru](https://mail.ru/dima.kukula.04@mail.ru)
* Learn.epam.com: [dima.kukula.04@mail.ru](https://mail.ru/dima.kukula.04@mail.ru)
* Autocode: [dima.kukula.04@mail.ru](https://mail.ru/dima.kukula.04@mail.ru)
* Gitlab: [dima.kukula.04@mail.ru](https://mail.ru/dima.kukula.04@mail.ru)
* Github: <https://github.com/DmitryKukula>

Глава 1 Теоретические основы ПРОГРАммирования по направлению .NEt C# development

* 1. Общая характеристика курса

Курс по направлению .net C# представляет собой сборник теории и практики по языку программирования. Во время его прохождения студент изучает синтаксис, логику и суть языка. За предложенное время предлагается изучить теорию от типов данных до потоков.

Также просмотрены лекции по профориентации:

1. Краткий обзор IT-профессий (00:20:30)
2. SAP Development & Consulting (1:07:20)
3. Data Analytics Engineering (1:09:35)
4. Functional Testing (1:20:48)
5. Robotic Process Automation (1:21:35)
6. Performance Optimization (1:09:39)
7. Cloud & DevOps (1:45:59)
   1. Модули курса

Закончены 14/18 модулей курса и изучена вся теория, предоставленная в них.

Изученная теория:

1. Value types (Built-in value types section): Типы значений и ссылочные типы — это две основные категории типов C#. Переменная типа значения содержит экземпляр типа. Это отличается от переменной ссылочного типа, которая содержит ссылку на экземпляр типа
2. Integral numeric types: Целочисленные числовые типы представляют целые числа. Все целочисленные числовые типы являются типами значений. Они также являются простыми типами и могут быть инициализированы литералами. Все целочисленные числовые типы поддерживают арифметические, побитовые логические операторы, операторы сравнения и равенства.
3. Constants: Константы — это неизменяемые значения, которые известны во время компиляции и не меняются в течение жизни программы.
4. Methods in C#: Метод — это блок кода, содержащий ряд операторов. Программа вызывает выполнение инструкций, вызывая метод и указывая все необходимые аргументы метода. В C# каждая выполняемая инструкция выполняется в контексте метода.
5. Named and optional arguments (Named arguments section): Именованные аргументы позволяют указать аргумент для параметра путем сопоставления аргумента с его именем, а не с его позицией в списке параметров. Необязательные аргументы позволяют опустить аргументы для некоторых параметров. Оба метода можно использовать с методами, индексаторами, конструкторами и делегатами.
6. Bool: Ключевое слово типа bool — это псевдоним структурного типа .NET System.Boolean, который представляет логическое значение, которое может быть либо истинным, либо ложным.
7. Boolean logical operators—AND, OR, NOT, XOR: Логические операторы выполняют логические операции с булевыми операндами. К операторам относятся унарное логическое отрицание (!), бинарное логическое И (&), ИЛИ (|) и исключающее ИЛИ (^), а также бинарное условное логическое И (&&) и ИЛИ (||).
8. C# operators and expressions (the section on operator precedence): C# предоставляет ряд операторов. Многие из них поддерживаются встроенными типами и позволяют выполнять основные операции со значениями этих типов. Эти операторы включают следующие группы: арифметические операторы, операторы сравнения, булевы операторы и другие
9. Selection statements: Операторы if, if-else и switch выбирают операторы для выполнения из многих возможных путей на основе значения выражения. Оператор if выполняет оператор только в том случае, если предоставленное логическое выражение оценивается как истинное. Оператор if-else позволяет выбрать, какой из двух путей кода следует использовать, на основе логического выражения. Оператор switch выбирает список операторов для выполнения на основе совпадения шаблона с выражением.
10. Pattern matching: Вы используете выражение is, оператор switch и выражение switch для сопоставления входного выражения с любым числом характеристик. C# поддерживает несколько шаблонов, включая объявление, тип, константу, реляционное значение, свойство, список, переменную и отбрасывание. Шаблоны можно комбинировать с помощью ключевых слов логической логики и, или, и не.
11. Floating-point numeric types: Числовые типы с плавающей запятой представляют действительные числа. Все числовые типы с плавающей запятой являются типами значений. Они также являются простыми типами и могут быть инициализированы литералами. Все числовые типы с плавающей запятой поддерживают арифметические операторы, операторы сравнения и операторы равенства.
12. Built-in numeric conversions: C# предоставляет набор целочисленных числовых типов и типов с плавающей запятой. Существует преобразование между любыми двумя числовыми типами, явное или неявное.
13. ?: operator - the ternary conditional operator: Условный оператор ?:, также известный как тернарный условный оператор, оценивает логическое выражение и возвращает результат одного из двух выражений в зависимости от того, является ли логическое выражение истинным или ложным.
14. Exceptions and Exception Handling: Функции обработки исключений языка C# помогают справляться с любыми непредвиденными или исключительными ситуациями, возникающими во время работы программы. При обработке исключений используются ключевые слова try, catch и finally, чтобы попытаться выполнить действия, которые могут оказаться неудачными, для обработки сбоев, когда вы решите, что это разумно, и для последующей очистки ресурсов.
15. Use exceptions
16. Exception Handling (C# Programming Guide)
17. Creating and Throwing Exceptions
18. ArgumentException Class
19. ArgumentNullException Class
20. ArgumentOutOfRangeException Class
21. The section on the while statement: Оператор while выполняет оператор или блок операторов, пока указанное логическое выражение оценивается как истинное. Поскольку это выражение вычисляется перед каждым выполнением цикла, цикл while выполняется ноль или более раз. Цикл while отличается от цикла do тем, что он выполняется один или несколько раз.
22. Jump statements: break и continue используются для прерывания или перехода к следующей итерации цикла
23. Arithmetic operators: инкрементные и декрементные операторы ++ и – помогают облегчить запись и функционал кода
24. The for statement: Оператор for выполняет оператор или блок операторов, пока указанное логическое выражение оценивается как истинное.

Глава 2 Практическая часть курса

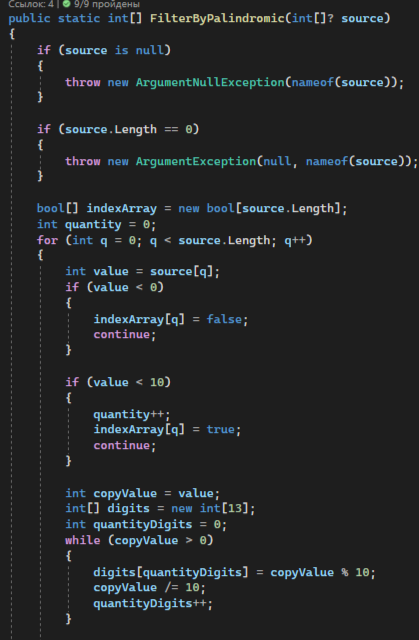
За период практики были пройдены все задачи и тесты курса:

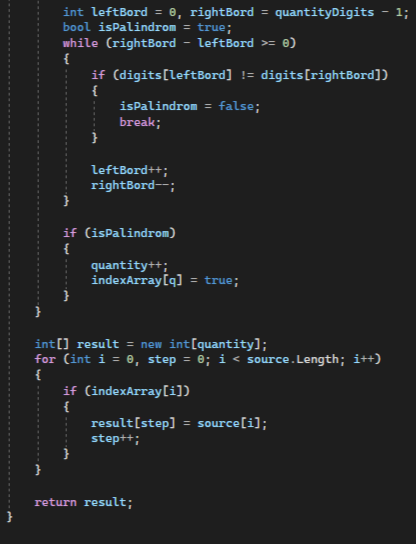
1. **Getting Started with Autocode**
2. **Integers (100)**
3. **Static Methods (100)**
4. **Bools & Logic Operators (100)**
5. **Conditional Statements (if) (35, потом задачу сдал на 100)**
6. **Real Numbers (100)**
7. **Conditional Statements (if-else) (100)**
8. **Exceptions (100)**
9. **Iteration Statements (while) (100)**
10. **Iteration Statements (for, do) (100)**
11. **Formatting and Parsing Strings (неполный)**
    1. Transform double (100)
    2. ISBN Validator (100)
12. **Working with Arrays (84)**
    1. Transformer in words (77)
    2. Anagrams (89)
    3. Spiral matrix (100)
    4. Rotate Matrix (70)
13. **Implementing Search Algorithms (100)**
    1. Find maximum (100)
    2. Balance element (100)
    3. Binary Search (100)
    4. Relocation elements (100)
    5. Filter by Palindromic (100)
    6. Filter by Digit (100)
14. **Implementing Sorting Algorithms (98)**
    1. Selection Sort (100)
    2. Insertion Sort (100)
    3. Bubble Sort (100)
    4. Quick Sort (90)
    5. Jagged Arrays (100)
15. **Solving Problems with Recursion (100)**
    1. Shift Array Elements (Recursion) (100)
    2. Getting a Character Index (Recursion) (100)
    3. Palindromic number (Recursion) (100)
    4. Looking for Chars (Recursion) (100)
    5. Looking for Array Elements (Recursion) (100)
    6. Find Maximum (Recursion) (100)
16. **Streams in .NET (100)**
    1. Working with Streams (100)

Во всех задач встретилось много интересных идей и реализация, многие из которых были подчеркнуты. Больше всех, для меня была интересна задача «Filter by Palindromic», т.к. в ходе её решения были придуманы 2 или 3 алгоритма. Алгоритмы не проходил один из тестов (очень долгая проверка). Методом проб и ошибок выявилось, что метод Array.Resize в цикле занимает очень много времени.

Задача: написать функцию, которая принимает массив целых чисел, как аргумент и возвращает новый массив, состоящий из чисел первого, которые являются палиндромами.

Решение:





Заключение

В результате прохождения этого курса был изучен новый язык С#, его синтаксис, особенности и получены базовые знания по возможностям. В общей сумме за время проведения практики было решено 33 задачи, 29 из них на 100 баллов. Прослушал все вебинары, узнал определённую новую информацию. Прошёл курс по контролю версий в Git, также выделил для себя некоторые моменты, которых не знал до этого.

Список использованных источников

1. <https://learn.microsoft.com/en-us/dotnet/csharp/> - документация по C# от Microsoft.
2. Herbert Schildt – The complete reference C# 4.0 – Издательский дом “Вильямс”, 2011.
3. Эндрю Троелсен – “Pro C# 10 with .NET 6” - Apress; 11st ed. Edition 2022