



УТВЕРЖДАЮ
Директор по персоналу ООО «Рубиус»
(по Доверенности от № 05 от 01.01.2022)
/ Ольга Сергеевна Мальцева

«26» июля 2022 г.

**Учебно-тематическое планирование
к программе повышения квалификации
«Основы программирования с использованием C#»**

Пояснительная записка

Актуальность программы. Данная программа является актуальной, так как язык C# является лидером по опросам среди программистов. Большинство участников опроса портала Geekbrains (90 %) — профессиональные разработчики с опытом более трёх лет. C# покорила сердца 60 % респондентов и занял восьмое место в десятке лидеров. По числу проектов на StackOverflow этот язык — третий.

Среди преимуществ C#:

- Простые паттерны асинхронного программирования;
- Работа приложений всюду, где есть .net framework — без перекомпиляции;
- Мобильная разработка под android, ios и windows phone с xamarin;
- Автоматическое управление памятью, сборщик мусора;
- Написание запросов с sql-подобным синтаксисом внутри кода c# (проект linq);
- Удобные фреймворки для создания mvс-приложений и работы с restful api;
- Хороший базовый набор библиотек;
- Статическая типизация — упрощает выявление ошибок.

Специфика данной программы обоснована тем, что Востребованность C# закономерна — он реализует многие преимущества C++ и Java, но по простоте сравним с Visual Basic. С помощью C# можно создавать игры на движке № 1 в мире — Unity, мобильные и веб-приложения, десктопные Windows-программы на каждый день и большие корпоративные продукты, в том числе банковские.

Около 34 % популярных бесплатных мобильных игр сделаны на Unity. На нем же работает большинство приложений виртуальной и дополненной реальности. Доля юнити-проектов среди игр для VR-шлемов и очков Samsung Gear достигает 90 %, для оборудования Oculus Rift — 53 %.

Данная программа повышения квалификации создает условия для **непрерывного образования** разработчиков посредством применения и совершенствования их навыков, знаний и умений на практике, в своём рабочем пространстве.

Новизна программы состоит в том, что все теоретические знания, полученные на курсе, закрепляются на практических примерах. Курс ведут опытные преподаватели, профессионалы своего дела. Все они успешно сочетают преподавание с практической работой.

Направленность программы: естественно-научная.

Данная программа **предназначена** для всех Си-программистов и на всех платформах. Она будет полезна как опытным, так и начинающим разработчикам на языке С, которые хотят развить свои профессиональные навыки.

Данная программа **рассчитана** на людей, имеющих в\о, ср.п\о.

Данная программа **рассчитана** на 75 академических часов (48 часов аудиторных занятий и 27 часов самостоятельной работы). Литературные ресурсы указаны ниже в списке учебной литературы.

Программа состоит из 10 модулей. Все вместе они дают целостное содержание, которое необходимо для профессионального становления разработчиков в данной сфере.

В результате освоения программы студенты получают удостоверение, подтверждающие повышение квалификации.

Целью программы является получения теоретических знаний и практических навыков, необходимых для работы С# программистом.

По итогам освоения программы слушатели смогут:

- Программировать на С#
- Управлять программными потоками
- Работать с данными
- Разрабатывать UI
- Работать с инструментами разработчика

Организационные формы учебной и познавательной деятельности

Для решения указанных задач особое внимание уделяется **организационным формам** учебного процесса, включающим инновационные социальные формы работы:

- аудиторные занятия под руководством тренера;

- обязательная самостоятельная работа по заданию преподавателя (домашняя работа, презентации, проведение исследования по Case Study и тд.);
- индивидуальная самостоятельная работа с Интернет-ресурсами.

Методы и принципы обучения

Методы обучения, реализуемые в рабочей программе, реализуемые в рамках индивидуального подхода к каждому студенту группы:

- Метод проблемно-ориентированного обучения
- Метод тренингов
- Метод кейсов

Данная рабочая программа реализуется на основе следующих **дидактических принципов** обучения:

- принцип мотивации – поддержание тренером мотивации обучения на высоком уровне, принимая за основу потребности учащихся.
- принцип лично-ориентированной направленности обучения – равновесное и равноправное взаимодействие всех участников учебного процесса, направленное на достижение общей цели;
- принцип профессиональной направленности – введение в содержание обучения профессионально значимого материала;
- принцип создания положительного отношения к учению – развитие сознательного отношения и мотивации к изучению программирования на протяжении всего периода обучения в центре и после окончания обучения;
- принцип моделирования профессиональной ситуации общения;
- принцип системности.

Учебный план

Тема	Количество часов
Модуль 1. Основы программирования <ol style="list-style-type: none">1. Введение в курс2. Общие сведения о вычислительной технике и программировании3. Обзор языков программирования4. Обзор платформы .Net и средств разработки5. Знакомство с языком C#6. Понятие сборки (Assembly) в .Net7. Установка и настройка Visual Studio	2,7 академ. часа
Модуль 2. Введение в язык C# <ol style="list-style-type: none">1. Создание и настройка Visual Studio2. Первая программа3. Запуск программы4. Введение в язык C#: синтаксис и ключевые слова5. Введение в язык C#: переменные и типы данных6. Введение в язык C#: пространства имён7. Введение в язык C#: комментарии8. Основы работы с Git и GitHub (commit, branch, pull, push)9. Переменные и типы данных: преобразование типов10. Ссылочные и значимые типы11. Nullable типы12. Неявно типизированные переменные13. Операции с числами14. Логические операции, операции сравнения	10,6 академ. часа

<ol style="list-style-type: none"> 15. Условные конструкции (if/else/switch) 16. Разделение логики: использование методов 17. Передача параметров по ссылке и значению, модификаторы ref и out 18. Массивы и многомерные массивы 19. Циклы и условия 20. Базовые алгоритмы сортировки массивов 21. Обобщённые коллекции: массивы, списки, словари 22. Строки. Работа со строками. 23. Работа с классом StringBuilder 24. Преобразование строк в примитивные типы (Parse, TryParse, Convert) 25. Обработка ошибок пользовательского ввода 26. Пользовательские типы: структуры и классы 27. Использование классов и структур 28. Члены класса (методы, поля) 29. Перечисления (enum) 30. Значения типов по умолчанию 31. Значение null 32. Nullable-типы 33. Преобразование типов: Упаковка и распаковка объектов, операции is (as) 34. Работа с датами и временем (класс System.DateTime) 35. Математические операции (класс System.Math) 36. Регулярные выражения (System.Regex) 	
<p>Модуль 3. Введение в ООП</p> <ol style="list-style-type: none"> 1. Обзор основных концепций ООП 2. Классы и объекты классов 3. Инкапсуляция, наследование и полиморфизм 	2,7 академ. часа

<ol style="list-style-type: none"> 4. Модификаторы доступа (public, private, protected) 5. Абстрактные классы и методы 6. Виртуальные методы. Перегрузка и сокрытие методов 7. Интерфейсы 8. Обзор типовых шаблонов проектирования 	
<p>Модуль 4. ООП в С#</p> <ol style="list-style-type: none"> 1. Иерархия классов .Net Framework (класс Object и переопределение его методов) 2. Обобщённые типы (Generics) 3. Конструкторы класса и наследование 4. Модификаторы const, readonly, static 5. Уровни доступности для классов. Модификатор internal и его применение 	2,7 академ. часа
<p>Модуль 5. Инструменты разработчика</p> <ol style="list-style-type: none"> 1. Отладка приложений – работа в debug-режиме 2. Повторное использование кода: NuGet 3. Обзор систем управления версиями 4. Профилирование приложений (основные сведения) 5. Тестирование приложений: понятия, юнит-тесты, TDD 	2,7 академ. часа
<p>Модуль 6. Делегаты и события</p> <ol style="list-style-type: none"> 1. Общие сведения о делегатах 2. Создание и работа с делегатами 3. События и событийно-управляемая модель приложения 	2,7 академ. часа

<p>Модуль 7. Расширенные возможности C#</p> <ol style="list-style-type: none"> 1. Обобщённые делегаты Action<> и Func<> 2. Анонимные методы 3. Лямбда-выражения 4. Локальные функции 5. LINQ 6. Концепция исключений 7. Обработка исключений (try, catch, finally) 8. Генерация собственных исключений 9. Расширяющие методы (Extension methods) и их использование 10. Частичные (partial) классы и методы 11. Рефлексия 12. Атрибуты 13. Работа с динамическими типами 14. Сборка мусора и освобождение ресурсов 	<p>5,3 академ. часов</p>
<p>Модуль 8. Многопоточность</p> <ol style="list-style-type: none"> 1. Введение в многопоточность (Thread, Task) 2. Синхронизация потоков 3. Параллельное программирование с использованием Task Parallel Library 4. Отмена выполнения параллельных задач 5. Async/Await 6. Обработка ошибок в асинхронных методах 7. Parallel.For и Parallel.Foreach 8. Parallel LINQ 9. Потокбезопасные коллекции 	<p>5,3 академ. часов</p>

Модуль 9. Работа с данными

1. Работа с локальными данными (File, FileInfo, Directory, DirectoryInfo, Path)
2. Работа с потоками данных (Streams)
3. Архивация и сжатие файлов
4. Сериализация и десериализация объектов
5. Понятие баз данных и СУБД
6. Модели данных, функции СУБД
7. Типовые архитектуры ИС
8. Сущности, атрибуты, связи
9. Реляционная модель данных
10. Язык SQL
11. Запросы через генерацию SQL
12. NoSQL. NoSQL vs реляционная модель
13. Object-Relational Mapping (ORM) и ORM-фреймворки
14. Общие сведения об Entity Framework Core
15. Code-first и Database-first подходы
16. Миграции
17. Создание и чтение файлов XML и JSON
18. Общие сведения о ASP.NET
19. Создание веб-приложений на платформе ASP .NET Core
20. Паттерн MVC (Model-View-Controller)

10,6 академ. часов

<p>Модуль 10. Разработка графического пользовательского</p> <ol style="list-style-type: none"> 1. Графический интерфейс: технологии разработки и история их развития 2. Общие сведения о паттерны проектирования UI: MVC, MVP, MVVM Windows Forms и WPF (Windows Presentation Foundation) 3. Разработка GUI с использованием WPF и паттерна MVVM 4. Использование XAML 5. Ресурсы и стили в WPF 6. Привязка к данным (Data Binding) в WPF 	2,7 академ. часа
<p>Самостоятельная работа</p>	27 академ. часов
<p>Итого</p>	75 академ. часов

Учебно-тематический план

№	Тема	Количество часов (академических)		
		Всего	Теория	Практика
1	Основы программирования	2	2,7	-
2	Введение в язык C#	10	10,6	2,7
3	Введение в ООП	4	2,7	2,7
4	ООП в C#	4	2,7	2,7
5	Инструменты разработчика	4	2,7	2,7
6	Делегаты и события	4	2,7	2,7
7	Расширенные возможности C#	6	5,3	2,7
8	Многопоточность	6	5,3	2,7
9	Работа с данными	14	10,6	5,4
10	Разработка графического пользовательского	4	2,7	2,7
	Контрольное занятие	-	-	-
	ВСЕГО	75	48	27

Текущий и итоговый контроль

Для эффективного управления и контроля за качеством усвоения учащимися учебного материала текущего контроля, которая в себе следующие элементы:

- *проверочные работы* (под контрольными работами и проверочными заданиями понимаются все виды работ, проводимые непосредственно в ходе аудиторных занятий и имеющие целью проверку состояния знаний учащихся; баллы, полученные за эти виды работ, включаются в качестве составной части либо в общую оценку за практические занятия (макс 50 баллов).
- *контрольный урок* (это тип проверочной работы, которая позволяет оценить качество усвоения учащимися содержания курса и уровень сформированности знаний, умений и навыков) проводится по завершении изучения курса в виде тестирования (максимум 50 баллов).

Итоговая проверка знаний осуществляется по средствам контрольного занятия. Итоговая оценка за курс представляет собой совокупность оценок за проверочные работы и контрольный тест.

Шкала оценивания представлена ниже:

<i>Проверочные работы №</i>	1	2	3	4	5	6	7	8	9	10	<i>Итого</i>
Максимальный балл	5	5	5	5	5	5	5	5	5	5	50

<i>Контрольный тест</i>	<i>Оценка модулей 1-5</i>			<i>Оценка модулей 6-10</i>			<i>Итого</i>
Максимальный балл	25			25			50

Итоговый балл: 100 баллов максимум

Результаты контрольного занятия и проверочных работ являются доказательством успешного\неуспешного прохождения курса\модуля и являются основанием к выдаче сертификата\диплома.

Список учебной литературы

1. C# 4.0 и платформа .NET 4 для профессионалов (+ CD-ROM) | Кристиан Нейгел, Билл Ивсен, Джей Глинн, Карли Уотсон, Морган Скиннер, 2011
2. C# 4.0. Полное руководство | Шилдт Герберт, 2013 Чистый код | Роберт Сесил Мартин, 2008
3. Идеальный программист | Роберт Сесил Мартин, 2011
4. Паттерны проектирования | Эрик Фримен, Элизабет Робсон, Берт Бейтс, Кэти Сьерра, 2014
5. Эффективное программирование на C#. 50 способов улучшения кода | Билл Вагнер, 2004 Перри, Грег
Программирование на C для начинающих | Грег Перри , 2015

