

ЯС.
Буд-т.
ЗК.

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

(название учреждения высшего образования)

УТВЕРЖДАЮ

Проректор по учебной работе
Белорусского государственного университета

А.Д. Толстик

(подпись)

(и О.Фамилия)

30
(дата утверждения)

Регистрационный № УД- 126 /уч.

ЯЗЫК C# AND ОСНОВЫ ПЛАТФОРМЫ .NET

(название учебной дисциплины)

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:**

1-31 03 01-01 Математика (научно-производственная деятельность)

(код специальности)

(наименование специальности)

2015 г.

Учебная программа составлена на основе ОСВО 1-31 03 01-01-2013 от 30.08.2013; УП для специальности 1-31 03 01-01 “Математика (научно-производственная деятельность)” № G31-140/уч. от 30.05.2013г.

СОСТАВИТЕЛИ:

А.И. Кравчук, доцент кафедры веб-технологий и компьютерного моделирования Белорусского государственного университета, кандидат физико-математических наук, доцент;

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой веб-технологий и компьютерного моделирования
(название кафедры – разработчика программы)

(протокол № 10 от 14.05.2015г.);

Учебно-методической комиссией механико-математического факультета БГУ
(название учреждения высшего образования)

(протокол № 6 от 26.05.2015г.).



ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

.NET Framework представляет собой программную платформу, выпущенную компанией Microsoft в 2002 году, для создания приложений на базе семейства операционных систем Windows, а также некоторых операционных систем производства не Microsoft, таких как Mac OS X и различные дистрибутивы Unix и Linux. Платформа .NET

- обеспечивает возможность взаимодействия с существующим программным кодом;
- позволяет поддерживать многочисленные языки программирования (C#, Visual Basic, F#, S# и т.д.);
- имеет общий исполняющий механизм, используемый всеми поддерживаемыми .NET языками;
- характеризуется полной интеграцией языков – поддерживается межъязыковое наследование, обработка исключений и отладка кода;
- имеет обширную библиотеку базовых классов, позволяющую избегать сложностей, связанных с выполнением прямых вызовов к API-интерфейсу, и предлагает согласованную объектную модель, которую могут использовать все поддерживаемые .NET языки.

Наличие перечисленных функциональных возможностей делает платформу .NET сверхмощным инструментом для создания различного рода приложений. Разработчики программного обеспечения на основе платформы .NET в настоящий момент являются востребованными, что делает платформу актуальной для изучения студентами.

Специальный курс «Язык CSharp и основы платформы .NET» имеет прикладную направленность.

Основными целями курса «Язык CSharp и основы платформы .NET» являются:

- дать теоретическую подготовку по языку C#;
- дать теоретическую подготовку по базовым технологиям .NET Framework;
- дать практические навыки создания программного обеспечения на примере разработки WPF-приложений;
- познакомить с принципами первичного тестирования разрабатываемого продукта;

Основными задачами курса «Язык CSharp и основы платформы .NET» являются:

- научить строить архитектуру разрабатываемого приложения в соответствии с принципами ООП и SOLID;
- научить пользоваться сторонними программами и инструментами в процессе разработки реальных программных продуктов.

Изучение дисциплины «Язык CSharp и основы платформы .NET» базируется на знаниях, полученных при изучении университетских курсов по информатике и программированию, в частности, таких современных языков программирования, как C++, Visual Basic, Java.

В результате изучения дисциплины студенты будут **знать:**

- синтаксис простых конструкций языка C#;
- основные концепции платформы .NET и ее базовой части Framework;

уметь:

- создавать и вызывать методы;
- перехватывать, обрабатывать и выбрасывать исключения;
- создавать и использовать новые типы (перечисления, классы и структуры) и понимать различия между ссылочными и значимыми типами данных;
- выполнять простые операции для взаимодействия с файловой системой;
- контролировать область видимости и время жизни членов типа;
- использовать наследование для разработки новых ссылочных типов данных;
- управлять временем жизни объектов и контролировать использование ресурсов;
- определять свойства и индексы для инкапсулирования данных и определять операторы для этих данных;
- отделять действие от метода, реализующего это действие, и использовать отделенные операции для обработки асинхронных событий;
- использовать коллекции для объединения данных и использовать обобщения для реализации типизированных классов, структур, интерфейсов и методов;
- реализовывать собственные классы коллекций, поддерживающие возможность перебора элементов;
- создавать запросы к данным, находящимся в памяти, с использованием LINQ;
- выполнять первичное тестирование проекта на этапах разработки;
- планировать и внедрять архитектуру проекта в соответствии с принципами ООП;

владеть:

- навыками работы в среде Microsoft Visual Studio 2012;

Курс «Язык CSharp и основы платформы .NET» рассчитан на 106 часов в V семестре, из них 54 аудиторных часа, в том числе 20 лекционных часов, 30 практических занятий и 4 часа контролируемой самостоятельной работы, которые проводятся на персональных компьютерах на базе компьютерных классов ММФ БГУ. Рекомендуется следующее распределение часов по видам учебной работы.

	Экзамен, семестр	Зачет, семестр	Всего часов	В том числе аудиторных	Из них		
					Лекций	Практических занятий	Контрольных работ
3 курс		5	106	54	20	30	4

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Тема 1. Введение в .NET Framework 4.

Модуль описывает цели .NET Framework 4 и возможности, доступные при создании приложений с помощью C# и Visual Studio 2010. Модуль описывает следующие темы:

- Знакомство с платформой .NET Framework.
- Сборки в .NET, локальные и глобальные сборки, GAC.
- Создание проектов с Visual Studio 2010. Типы проектов VS 2010.
- Написание приложений на C#.
- ILDASM, .NET Reflector.
- Построение графических приложений.
- Отладка исходного кода, документирование приложения.

Тема 2. Основные программные конструкции C#.

Модуль описывает основные типы данных и программные конструкции языка C#, их синтаксис и семантику. Модуль описывает следующие темы:

- Объявление переменных и присваивание значений.
- Использование выражений и операций языка в C#.
- Использование операторов выбора.
- Использование операторов цикла.
- Создание и использование массивов.

Тема 3. Объявление и вызов методов в C#.

Модуль представляет методы, описывает их определение и использование. Модуль описывает следующие темы:

- Определение и вызов методов.
- Спецификация необязательных и выходных параметров.
- Взаимодействие типов, объектов, стека потока и управляемой кучи во время выполнения.

Тема 4. Создание новых типов в C#.

Модуль объясняет, как создавать собственные типы и описывает различия между ссылочными типами (reference type) и типами значений (value type). Модуль описывает следующие темы:

- Создание и использование перечислений.
- Создание и использование классов.
- Создание и использование структур.
- Сравнение ссылочных типов с типами значений.

Тема 5. Инкапсуляция данных и методов.

Модуль описывает, как использовать некоторые из модификаторов доступа, предоставляемых C#, для реализации принципа инкапсуляции. Кроме того модуль также вводит модификатор static, который позволяет определить члены типа, которые могут быть разделяться несколькими экземплярами одного и того же типа. Модуль описывает следующие темы:

3.1	Определение и вызов методов. Спецификация необязательных и выходных параметров.	1				2						
3.2	Взаимодействие типов, объектов, стека потока и управляемой кучи во время выполнения.	1										
4	Тема: Создание новых типов в C#	2				4				[1-3,4,5]	Опрос, отчеты по практическим упражнениям	
4.1-4.2	Создание и использование перечислений. Создание и использование классов. Создание и использование структур.	2				4						
5	Тема: Инкапсуляция данных и методов. Перегрузка операций	2				4				[1-3]	Опрос	
5.1-5.2	Управление видимостью членов типа. Свойства и индексы. Статические методы и данные. Перегрузка операций	2				4						
6	Тема: Наследование. Интерфейсы и абстрактные классы	2				6		2		[1-3,7-12]	Опрос	
6.1	Использование наследования для определения новых ссылочных типов.	1				3						
6.2	Определение и реализация интерфейсов. Определение абстрактных классов.	1				3						
7	Тема: Обработка исключений.	1								[1-3]	Опрос, отчеты по практическим упражнениям	
7.1-7.2	Обработка исключений. Возникновение исключений.	1										
8	Тема: Управление ресурсами в .NET	1								[3,9]	Опрос	
8.1-8.2	Сборка мусора. Введение в сборку мусора. Управление ресурсами	1										
9	Тема: Работа с файловой системой	1				2				[1-3]	Опрос, отчеты по практическим упражнениям	

Требования к курсовой работе

Курсовая работа – это творческое и самостоятельное исследование актуальных проблем определенного раздела по той или иной отрасли математики или специальному предмету, выполненное студентом самостоятельно.

Образовательными целями, достигаемыми при выполнении курсовой работы, являются приобретение, развитие и закрепление студентом следующих навыков и компетенций

- самостоятельной работы с математическими объектами,
- умения работать самостоятельно с учебной и научной литературой,
- умении создать осмысленный текст, содержащий, в частности, собственные суждения автора курсовой работы по избранной тематике.

Работа должна носить творческий характер, так как курсовая работа – это не только данные, скопированные из различных источников. Студент должен обязательно привнести в работу элементы самостоятельного исследования.

Темы курсовых работ предлагаются и утверждаются кафедрами. Студенты самостоятельно выбирают тему курсовой. Студент также может предложить свою тему курсовой работы, но обосновав при этом целесообразность ее разработки. Темы курсовых работ, выполняемых студентами за весь период обучения, желательно подбирать таким образом, чтобы они вместе с дипломным проектом составляли единую систему последовательно усложняемых и взаимосвязанных исследований.

Для формирования у обучающихся умений и навыков работы в команде возможна выдача группового задания, предусматривающего работу нескольких обучающихся над одной курсовой работой. В этом случае каждому из них должен быть установлен индивидуальный объем задач в соответствии с объемом и уровнем общих требований.

Задание по курсовым работам должно быть выдано обучающемуся, осваивающему содержание образовательной программы высшего образования I ступени: в очной форме получения высшего образования в первые две недели после начала семестра, в котором учебными планами они предусмотрены; в заочной форме получения высшего образования во время лабораторно-экзаменационной (установочной) сессии, предшествующей семестру, в котором учебными планами они предусмотрены.

Задание по курсовой работе выдается индивидуально и должно содержать конкретное название темы, необходимые исходные данные, перечень основных литературных источников. В задании указывается дата выдачи задания и представления работы к защите, задание подписывается студентом и руководителем работы.

Курсовая работа включает: титульный лист; задание на курсовую работу (курсовой проект); оглавление; перечень условных обозначений, символов и терминов (при необходимости); введение; основная часть; заключение; список использованной литературы; приложения.

Форма отчетности: студент предоставляет руководителю текст курсовой работы, оформляемый по следующим правилам. Курсовая работа печатается с использованием компьютера и принтера на одной стороне листа белой бу-

маги формата А4 (210x297 мм). Количество знаков в строке должно составлять 60-70, межстрочный интервал должен составлять 18 пунктов, количество текстовых строк на странице – 39-40. Устанавливаются следующие размеры полей: верхнего и нижнего – 20 мм, левого – 30 мм, правого – 10 мм. Объем курсовой работы, как правило, должен составлять 15-30 страниц. Иллюстрации, таблицы, список использованных источников и приложения при подсчете объема работ не учитываются. Основной текст курсовой работы должен иметь четкую и логичную структуру, содержащую оглавление, введение, основную часть, список использованных источников. Весь материал должен быть доказуемым с чётким изложением фактов и аргументов. Основные выводы и результаты должны быть сформулированы и обоснованы в соответствии с принятыми в математике правилами. В тексте должны быть правильно оформлены библиографические ссылки на использованные источники.

Обучающийся обязан представить руководителю курсовую работу, в срок, установленный заданием на курсовую работу, для проведения первичного анализа и принятия решения о допуске к защите. Руководитель курсовой работы оценивает качество ее выполнения, желательно публичное проведение процедуры защиты курсовой работы. При защите работы студент учится не только правильно излагать свои мысли, но и аргументировано отстаивать, защищать выдвигаемые выводы и решения.

В соответствии с учебным планом специальности на выполнение курсовой работы отводится 40 часов.

Примерная тематика курсовых работ

1. Разработка ASP.NET MVC приложения с адаптивным и отзывчивым дизайном.
2. Создание SPA-приложений с использованием ASP.NET Web API и Angular.js.
3. Создание SPA-приложений с использованием ASP.NET Web API и Knockout.js.
4. Внутреннее устройство типов .NET Framework.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

ЛИТЕРАТУРА

Основная:

1. Andrew Troelsen. Pro C# 5.0 and the .NET 4.5 Framework. Sixth Edition. Apress – 2012, 1487 p.
2. Albahari J., Albahari B. C# 5.0 in a Nutshell. Fifth Edition. O'Reilly Media – 2012, 1062 p.
3. Jeffrey Richter. CLR via CSharp. 3rd Edition. Microsoft Press – 2010, 896 p.

Дополнительная:

4. Эндрю Троелсен. Язык программирования C# 2010 и платформа .NET 4. 5-е издание. Apress – 2011, 1392 с.
5. Нейгел, Кристиан, Ивсен, Билл, Глинн, Джей, Уотсон, Карли. C# 4.0 и платформа .NET 4 для профессионалов.: Пер. с англ. – М. : ООО «И.Д. Вильяме», 2011. – 1440 с.
6. Л. Пауэре, М. Снелл. Microsoft Visual Studio 2008. СПб.: БХВ-Петербург, 2009. – 1200 с.
7. <http://msdn.microsoft.com/en-us/library>.
8. <http://msdn.microsoft.com/ru-ru/library/ms123401>.
9. <http://www.rsdn.ru>.
10. Алекс Макки. Введение в .NET и Visual Studio 2010 для профессионалов.: Пер. с англ. – М. : ООО «И.Д. Вильяме», 2011. – 416 с.
11. Jesse Liberty, Brian MacDonald. Learning C# 3.0: Master the fundamentals of C# 3.0. O'Reilly Media. – 2008, 704 p.
12. Judith Bishop. C# 3.0 Design Patterns. O'Reilly Media. – 2007, 292 p.

ПЕРЕЧЕНЬ ЗАДАНИЙ И КОНТРОЛЬНЫХ МЕРОПРИЯТИЙ УПРАВЛЯЕМОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Управляемая самостоятельная работа (УСР) обучающихся – это самостоятельная работа, выполняемая по заданию и при методическом руководстве и контролируемая на определенном этапе обучения преподавателем.

Целью УСР дополнительно к целям СР является целенаправленное обучение основным навыкам и умениям для выполнения СР.

УСР, как важная составная часть учебного процесса, в том числе по дисциплине «Язык CSharp и основы платформы .NET», должна обеспечиваться мотивацией, доступностью и качеством научно-методического и материально-технического обеспечения образовательного процесса, сопровождаться эффективной системой контроля и способствовать усилению практической направленности обучения. При выполнении УСР должны быть созданы условия, при которых обеспечивалась бы активная роль обучающихся в самостоятельном получении знаний и систематическом применении их на практике. Управление самостоятельной работой обучающихся должно осуществляться через разработку научно-методического обеспечения СР и осуществление контрольных мероприятий.

К **организационным формам** проведения УСР по дисциплине «Язык CSharp и основы платформы .NET» можно отнести аудиторную деятельность на практических (семинарских) занятиях. **Видами отчетности** УСР являются: контрольные работы, коллоквиумы, отчеты по практическим работам, другие.

Контроль УСР по дисциплине «Язык CSharp и основы платформы .NET» проводится преподавателем, как правило, во время аудиторных занятий и осуществляется в виде:

- экспресс-опроса на аудиторных занятиях;
- защиты практических заданий.

Учет результатов контроля текущей успеваемости студентов ведется преподавателем. Полученные студентом количественные результаты УСР учитываются как составная часть итоговой оценки по дисциплине в рамках рейтинговой системы.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ ЗАДАНИЙ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Индивидуальные задания для самостоятельной работы включают разработку приложений различного уровня сложности.

Задание 1.

- Разработать тип, позволяющий вычислять корень n -ой степени из числа методом Ньютона с заданной точностью.
- Предусмотреть возможность сравнения полученного результата со значе-

нием, рассчитываемым с помощью метода `Math.Pow` библиотеки классов `.NET Framework`.

- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 2.

- Разработать тип, позволяющий конвертировать неотрицательное десятичное значение целого числа в строку, содержащую двоичное представление этого значения.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 3.

- Разработать тип, реализующий алгоритм Евклида для вычисления НОД двух целых чисел (http://en.wikipedia.org/wiki/Euclidean_algorithm). Метод должен принимать выходной параметр, содержащий значение времени, необходимое для выполнения расчета. Добавить к разработанному типу дополнительную функциональность в виде перегруженных методов вычисления НОД для трех и т.д. целых чисел.
- Добавить к разработанному типу метод, реализующий алгоритм Стейна (бинарный алгоритм Эвклида) для расчета НОД двух целых чисел (http://en.wikipedia.org/wiki/Binary_GCD_algorithm). Метод должен принимать выходной параметр, содержащий значение времени, необходимое для выполнения расчетов. Добавить к разработанному типу дополнительную функциональность в виде перегруженных методов вычисления НОД для трех и т.д. целых чисел.
- Разработать метод, позволяющий сравнивать результаты работы разработанных методов на одинаковых наборах данных. (Разработать метод для отображения результатов работы с использованием гистограммы. Метод должен принимать в качестве двух обязательных параметров время выполнения, и в качестве необязательных параметров – ориентацию гистограммы (вертикальная или горизонтальная) и цвет для отображения ее полос).
- Создать unit-тесты для тестирования разработанных методов.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 4.

- Разработать класс «вектор» для работы с трехмерными векторами.
- Перегрузить для класса операции, допустимые для работы с трехмерными векторами.
- Создать unit-тесты для тестирования разработанных методов.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 5.

- Разработать класс «многочлен» для работы с многочленами от одной пе-

ременной.

- Перегрузить для класса операции, допустимые для работы с многочленами.
- Создать unit-тесты для тестирования разработанных методов.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 6.

- Разработать тип для работы с матрицами.
- Реализовать методы, позволяющие выполнять основные операции над матрицами, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение.
- В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.
- Создать unit-тесты для тестирования методов разработанного типа.

СРЕДСТВА ДИАГНОСТИКИ РЕЗУЛЬТАТОВ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ

Рекомендуются следующие формы диагностики компетенций:

1. Опрос.
2. Отчеты по практическим упражнениям с их устной защитой.
3. Электронные тесты.
4. Зачет.

**ПРОТОКОЛ
СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ ПО ИЗУЧАЕМОЙ
УЧЕБНОЙ ДИСЦИПЛИНЕ С ДРУГИМИ ДИСЦИПЛИНАМИ
СПЕЦИАЛЬНОСТИ**

Название дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Методы программирования и информатика	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол №10 от 14.05.2015г.)

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ

на 2016/2017 учебный год

№ п/п	Дополнения и изменения	Основание
1	Изъять из списка литературы (дополнительное) № 4, 6.	Четвертое

Учебная программа пересмотрена и одобрена на заседании кафедры Веб-технологий и компьютерного моделирования (протокол № 9 от 27.05 2016 г.)

Заведующий кафедрой

канд. физ.-мат. наук, доцент
(ученая степень, ученое звание)


(подпись)

В.С. Романчик
(И.О.Фамилия)

УТВЕРЖДАЮ

Зам. Декан факультета

канд. физ.-мат. наук, доцент
(ученая степень, ученое звание)

(подпись)

Д.Г. Медведев
(И.О.Фамилия)