

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кубанский государственный университет»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

Иванов А.Г.

« 29 »

мая

2015г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.ДВ.03.01 Основы разработки кросс-платформенных приложений

(код и наименование дисциплины в соответствии с учебным планом)

Направление подготовки/специальность 02.03.03 «Математическое
обеспечение и администрирование информационных систем»
(код и наименование направления подготовки/специальности)

Направленность (профиль) / специализация «Технология программирования»
(наименование направленности (профиля) специализации)

Программа подготовки академическая
(академическая /прикладная)

Форма обучения очная
(очная, очно-заочная, заочная)

Квалификация (степень) выпускника бакалавр
(бакалавр, магистр, специалист)

Краснодар 2015

Рабочая программа дисциплины «Основы разработки кросс-платформенных приложений» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 «Математическое обеспечение и администрирование информационных систем»

Программу составил(и):

А.А. Полупанов, доцент кафедры информационных технологий КубГУ, канд. техн. наук, доцент



Рабочая программа дисциплины «Основы разработки кросс-платформенных приложений» утверждена на заседании кафедры информационных технологий, протокол № 8 «29» апреля 2015 г.

Заведующий кафедрой (разработчика) Кольцов Ю.В.



Рабочая программа обсуждена на заседании кафедры интеллектуальных информационных систем, протокол № 5 «15» апреля 2015 г.

Заведующий кафедрой (выпускающей) Костенко К.И.



Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики, протокол № 5 «29» апреля 2015 г.

Председатель УМК факультета Малыхин К.В.



Рецензенты:

Рубцов С.Е., доцент кафедры математического моделирования ФГБОУ ВО «КубГУ»

Бегларян М.Е., заведующий кафедрой СГЕНД СКФ ФГБОУ ВО «Российский государственный университет правосудия»

1 Цели и задачи изучения дисциплины

1.1 Цель освоения дисциплины

Освоение основ программирования на платформе .NET Framework; изучения языка программирования C#; совершенствование навыков объектно-ориентированного программирования в рамках изучения C#.

Воспитательная цель: формирование свободного и творческого подхода к программированию на современных языках высокого уровня, интереса к наблюдению за тенденциями и новостями в области средств разработки программного обеспечения.

1.2 Задачи дисциплины

Основные задачи курса на основе системного подхода:

- введение в основные идеи повторного использования кода и компонентов приложения, проблемы коллективной разработки приложений;
- знакомство с платформой .NET Framework и сравнение её с другими платформами разработки ПО для ОС Microsoft Windows;
- изучение основных концепций платформы .NET Framework и её составных частей: CLR, IL, CTS, CLS, сборки, манифесты;
- изучение C# как языкового средства, наиболее полно отражающего возможности .NET Framework;
- формирование навыков создания приложений на языке C#;
- совершенствование и углубление навыков объектно-ориентированного программирования, изучение последних нововведений в области ООП, реализованных в C#;
- знакомство с основами создания приложений для взаимодействия с базами данных на основе технологии ADO.NET;
- изучение возможностей создания Web-приложений, серверная часть логики которых написана на языке C#;
- обзор альтернативных подходов к разработке приложений в рамках .NET Framework: WPF, WCF, WF.

Отбор материала основывается на необходимости ознакомить студентов со следующей современной научной информацией:

- об особенностях и последних достижениях в области разработки ПО на платформе .NET Framework;
- о положительных и отрицательных чертах подхода к программированию, реализованному в языке C#.

Содержательное наполнение дисциплины обусловлено общими задачами в подготовке бакалавра.

Научной основой для построения программы данной дисциплины является теоретико-прагматический подход в обучении.

1.3 Место дисциплины в структуре образовательной программы

Дисциплина «Основы разработки кросс-платформенных приложений» относится к вариативной части Блок 1. Дисциплины (модули).

Дисциплина «Основы разработки кросс-платформенных приложений» логически и содержательно-методически связана с такими дисциплинами как: «Разработка приложений в MS Visual Studio», «Администрирование в Linux», «Программирование и администрирование в Oracle», «WEB-программирование».

Входными знаниями для освоения данной дисциплины являются знания, умения и опыт, накопленный студентами в процессе изучения дисциплин «Фундаментальные дискретные модели», «Разработка приложений в MS Visual Studio».

Обучающийся должен:

- иметь базовые навыки в написании программ на процедурных и объектно-ориентированных языках;
- знать принципы создания и организации работы приложений в ОС MS Windows;
- быть знакомым с наиболее часто встречающимися структурами данных, уметь ими пользоваться и знать внутреннюю организацию.

1.4 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих профессиональных компетенций (ПК):

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	уметь	владеть
1.	ПК-5	готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ	основные идеи повторного использования кода и компонентов приложения, проблемы коллективной разработки приложений, современные системные программные средства: операционные системы, операционные и сетевые оболочки, сервисные программы; принципы работы платформы .NET Framework и её составные части: CLR, IL, CTS, CLS, GC, assemblies; основные концепции, элементы синтаксической и семантической организации, методов	в рамках профессиональной деятельности применять язык программирования C#, использовать современные системные программные средства: операционные системы, операционные и сетевые оболочки, сервисные программы; работать в среде программирования Microsoft Visual Studio; реализовывать алгоритмические решения на практике, в рамках бизнес-процессов	навыками современного объектно-ориентированного программирования; идеями и средствами коллективной разработки приложений, создания повторно-используемого кода; методологией управления качеством производственной деятельности, связанной с использованием современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ

2 Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 2 зач.ед. (72 часа), их распределение по видам работ представлено в таблице (для студентов ОФО).

Вид учебной работы		Всего часов	Семестры (часы)			
			8			
Контактная работа, в том числе:						
Аудиторные занятия (всего):						
Занятия лекционного типа		16	16	-	-	-
Лабораторные занятия		32	32	-	-	-
Занятия семинарского типа (семинары, практические занятия)		-	-	-	-	-
		-	-	-	-	-
Иная контактная работа:						
Контроль самостоятельной работы (КСР)		2	2			
Промежуточная аттестация (ИКР)		0,2	0,2			
Самостоятельная работа, в том числе:						
Курсовая работа		-	-	-	-	-
Проработка учебного (теоретического) материала		8	8	-	-	-
Выполнение индивидуальных заданий		10	10	-	-	-
Реферат		-	-	-	-	-
Подготовка к текущему контролю		3,8	3,8	-	-	-
Контроль:						
Подготовка к экзамену						
Общая трудоёмкость	час.	72	72	-	-	-
	в том числе контактная работа	50,2	50,2			
	зач. ед.	2	2			

2.2 Структура дисциплины

Распределение видов учебной работы и их трудоёмкости по разделам дисциплины.

Разделы дисциплины, изучаемые в 8 семестре (очная форма)

№ раздела	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1	Введение в .NET Framework	8	2	—	4	2
2	Архитектура .NET Framework	8	2	—	4	2
3	Введение в C#	8	2	—	4	2
4	Объектно-ориентированное программирование в C#	10	2	—	6	2
5	Продвинутое возможности языка C#	10	2	—	4	4
6	Windows Forms и ADO. NET	10	2	—	4	4
7	Web-приложения	12	4	—	4	4
8	Обзор изученного материала и	3,8			2	1,8

	прием зачета					
	Контроль самостоятельной работы (КСР)	2				
	Промежуточная аттестация (ИКР)	0,2				
	Итого по дисциплине:	72	16	–	32	21,8

Примечание: Л – лекционные занятия, ПЗ – практические занятия / семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

2.3 Содержание разделов дисциплины

2.3.1 Занятия лекционного типа

№ раздела	Наименование раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
1	Введение в .NET Framework	Знакомство с понятием программной платформы. Основные средства разработки приложений в MS Windows, их сильные и слабые черты. Повторное использование кода. Преимущества и недостатки .NET Framework	ЛР, задачи
2	Архитектура .NET Framework	Общая архитектура .NET. CLR. Схема трансляции. IL. Межязыковое взаимодействие. Общая система типов CTS. Общая спецификация языка CLS. Сборка мусора. Домены приложений. Обработка ошибок с помощью исключений. Сборки и манифесты. Классы .NET Framework. Создание приложений .NET. WPF, WCF, WF. Версии .NET	ЛР, задачи
3	Введение в C#	Структура приложения. Типы, переменные. Object. Управление потоком выполнения. Приведение типов	ЛР, задачи
4	Объектно-ориентированное программирование в C#	Объекты и структуры C#. Данные-члены: поля, события. Функции-члены: методы, свойства, конструкторы, финализаторы, операции, индексаторы. Типы аргументов метода. Статические члены. Переопределение методов. Наследование. Соккрытие. Абстрактные методы. Модификаторы. Интерфейсы. Generic-обобщения и ограничения	ЛР, задачи
5	Продвинутые возможности языка C#	Делегаты. Анонимные методы. Лямбда-выражения. События. _Классы коллекций в C#. Массивы.LINQ. Обработка исключений. Рефлексия	ЛР, задачи
6	Windows Forms и ADO. NET	Создание оконных приложений в C#. Формы. Контроллы. Основные элементы управления	ЛР, задачи

		и их свойства: Button, Checkbox, Radiobutton, Textbox, Label, списочные элементы управления. ADO. NET. Работа с данными. Локальные наборы данных. Генерация схемы с помощью XSD. Привязки к элементам формы	
7	Web-приложения	Обзор технологии ASP.NET. Архитектура приложения ASP.NET. Создание простейшего сайта. Web элементы управления. Работа с данными	ЛР, задачи

2.3.2 Занятия семинарского типа

Занятия семинарского типа учебным планом не предусмотрены.

2.3.3 Лабораторные занятия

№ работы	Наименование лабораторных работ	Форма текущего контроля
1	2	3
1	Введение в .NET Framework. Создание нового проекта в среде Microsoft Visual Studio 2013	Отчёт по ЛР
2	Классы, методы, свойства, индексаторы в C#	Отчёт по ЛР
3	Конструкторы, поля только для чтения, вызов конструкторов. Делегаты и события в C#	Отчёт по ЛР
4	Разработка Windows-приложения на языке C#. Простейшие графические возможности	Отчёт по ЛР
5	Создание простейших меню. Свойства и методы стандартных элементов управления класса Control, диалоговые окна класса CommonDialog	Отчёт по ЛР
6	Работа с массивами и строками в языке C#. Интерфейсы и коллекции	Отчёт по ЛР
7	Класс DataSet и доступ к данным в формате XML	Отчёт по ЛР

2.3.4 Примерная тематика курсовых работ (проектов)

Учебным планом не предусмотрены.

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Решение индивидуальных задач	Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие/ Ю.В.Кольцов [и др.]. – Краснодар: Кубанский гос. ун-т, 2015.- 111с., утвержденные кафедрой информационных технологий, протокол № 7 от 09.04.2015 г.

2	Отчёт по лабораторной работе	Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие/ Ю.В.Кольцов [и др.]. – Краснодар: Кубанский гос. ун-т, 2015.- 111с., утвержденные кафедрой информационных технологий, протокол № 7 от 09.04.2015 г.
---	------------------------------	--

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

– Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

– Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

– Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

– Информационно-коммуникационные технологии (ИКТ) – расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

– Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

– Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

– Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

– Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

– Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

– Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

– Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

– работа в малых группах (команде) - совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

– проектная технология - индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	количество интерактивных часов
8	Л, ЛР	Занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент»	12
Итого			12

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные средства для текущего контроля успеваемости и итоговой аттестации

4.1 Фонд оценочных средств для проведения текущего контроля

Перечень заданий текущего контроля

Перечень компетенций, проверяемых оценочным средством: ПК-5 – готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ.

- 1) Введение в .NET Framework. Создание нового проекта в среде Microsoft Visual Studio 2013.
- 2) Классы, методы, свойства, индексы в C#.
- 3) Конструкторы, поля только для чтения, вызов конструкторов. Делегаты и события в C#.
- 4) Разработка Windows-приложения на языке C#. Простейшие графические возможности.
- 5) Создание простейших меню. Свойства и методы стандартных элементов управления класса Control, диалоговые окна класса CommonDialog.
- 6) Работа с массивами и строками в языке C#. Интерфейсы и коллекции.
- 7) Класс DataSet и доступ к данным в формате XML.

Перечень индивидуальных задач текущего контроля

Перечень компетенций, проверяемых оценочным средством: ПК-5 – готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ.

Задача 1. Написать windows-приложение, которое: заполняет случайными целыми числами массив 5x10 из диапазона [0, 9], при помощи класса DataSet. Воспользовавшись вспомогательным массивом A[5], прибавить к каждому элементу исходного массива минимальный элемент каждой его строки. Заменить исходный массив новыми значениями по нажатию кнопки «Решить».

Задача 2. Написать windows-приложение, которое: заполняет случайными целыми числами массив 5x10 из диапазона [-9, 9], при помощи класса DataSet. Воспользовавшись вспомогательным массивом A[5], определить количество отрицательных элементов в каждой строке. Задачу выполнить по нажатию кнопки «Решить».

Задача 3. Написать windows-приложение, которое: заполняет случайными целыми числами массив 5x10 из диапазона [-9, 9], при помощи класса DataSet. Воспользовавшись вспомогательным массивом A[10], определить среднее арифметическое элементов каждого столбца. Задачу выполнить по нажатию кнопки «Решить», а сами значения вывести в новую строку, при помощи класса DataSet.

Задача 4. Написать windows-приложение, которое: заполняет случайными целыми числами массив 5x10 из диапазона [-3, 3], при помощи класса DataSet. Воспользовавшись вспомогательным массивом A[10], определить сумму элементов каждого столбца. Задачу выполнить по нажатию кнопки «Решить», а сами значения вывести в новую строку, при помощи класса DataSet.

Задача 5. Написать windows-приложение, которое: заполняет случайными целыми числами массив 5×10 из диапазона $[-9, 9]$, при помощи класса DataSet. Воспользовавшись вспомогательным массивом A[10], определить количество отрицательных элементов каждого столбца. Задачу выполнить по нажатию кнопки «Решить», а сами значения вывести в новую строку, при помощи класса DataSet.

Задача 6. Написать windows-приложение, которое: заполняет случайными целыми числами массив 5×10 из диапазона $[-9, 9]$, при помощи класса DataSet. Воспользовавшись вспомогательным массивом A[10], среднее арифметическое значение элементов каждого столбца. Задачу выполнить по нажатию кнопки «Решить», а сами значения вывести в новую строку, при помощи класса DataSet.

Задача 7. Построить Windows-приложение «Арифметические операции». В одно или два текстовых окна пользователь вводит значения. По нажатию командной кнопки, задающей тип арифметической операции, над введенными значениями выполняется соответствующая операция, и её результат выводится в текстовое окно, предназначенное для вывода значений.

Задача 8. Построить Windows-приложение «Графические примитивы». В главное окно, которого выводятся графические объекты: круг, эллипс, линия.

Задача 9. Построить Windows-приложение «Графические примитивы». В главное окно, которого выводятся графические объекты: квадрат, треугольник, пунктирная линия.

Задача 10. Построить Windows-приложение «Графические примитивы». В главное окно, которого выводятся графические объекты: прямоугольник, пятиугольник, ломаная линия.

Задача 11. Найти количество положительных и сумму нечётных элементов массива B[15].

Задача 12. Найти сумму положительных и количество нечётных элементов массива A[10].

Задача 13. Вычислить среднее арифметическое элементов массива T[15], удовлетворяющих условию $5 \leq T[i] \leq 15$.

Задача 14. Вычислить среднее геометрическое четных и сумму нечётных элементов массива C[10].

Задача 15. Найти количество элементов массива B[16], кратных 4 и не больше заданного числа a.

Задача 16. Найти сумму элементов одномерного массива размером 5. Разделить каждый элемент исходного массива на полученное значение. Результат сохранить в том же массиве. Напечатать в одной строке.

Задача 17. Найти среднее значение элементов заданного массива размером 6. Преобразовать исходный массив, вычитая из каждого элемента среднее значение.

Задача 18. Вычислить длину вектора X размером 7.

Задача 19. Определить среднее значение элементов массива. Затем найти индекс элемента массива, наиболее близкого к среднему значению.

Задача 20. Задан массив размером 10. Если сумма элементов окажется больше 10, то найти количество четных элементов, иначе – произведение нечётных.

Задача 21. Задан массив размером 10. Если произведение элементов окажется больше 100, то найти сумму положительных элементов, иначе – количество отрицательных.

Задача 22. Задан массив размером 10. Если количество чётных элементов окажется больше 5, то подсчитать количество положительных элементов, иначе – сумму нечётных.

4.2 Фонд оценочных средств для проведения итоговой аттестации

Зачетно-экзаменационные материалы для промежуточной аттестации (зачёт)

Список задач к промежуточной аттестации

Перечень компетенций, проверяемых оценочным средством: ПК-5 – готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ.

- 1) Вводится последовательность целых чисел, 0 – конец последовательности. Определить сумму тех из них, порядковые номера которых являются числами Фибоначчи. Вывести на экран сумму, а также количество элементов ряда Фибоначчи.
- 2) Вводится последовательность из N целых чисел. Определить, является ли последовательность знакопеременной.
- 3) Вводится последовательность целых чисел, 0 – конец последовательности. Содержит ли последовательность хотя бы три отрицательных числа.
- 4) Дан массив целых чисел A[5], элементы которого создаются при помощи случайной генерации [1; 9]. Найти минимальный элемент массива, максимальный элемент массива, поменять их местами и вывести полученный массив на экран.
- 5) Дан массив целых чисел A[10], элементы которого создаются при помощи случайной генерации [-99; 99] исключая подмножество [-9; 9]. Если последняя цифра в каждом элементе массива не равна 0, то заменить её на 0 и вывести на экран полученный массив.
- 6) Дан массив целых чисел A[10], элементы которого создаются при помощи случайной генерации [0; 9]. Найти индексы тех элементов, которые больше своего левого соседа, и количество таких элементов. Найденные индексы выводить в порядке их убывания.
- 7) Дана матрица целых чисел A[4][4], элементы которой создаются при помощи случайной генерации из диапазона [0; 4]. Сформировать новый массив B[4], содержащий суммы каждого столбца матрицы A. Вывести оба массива на экран.
- 8) Дана матрица целых чисел A[8][8], элементы которой создаются при помощи случайной генерации [0; 9]. Поменять местами элементы двух заданных столбцов и вывести полученную матрицу на экран.
- 9) Дана бинарная матрица A[4][4], элементы которой создаются при помощи случайной генерации из диапазона [0; 1]. Сформировать новый массив B[4], содержащий количество 1 каждой строки матрицы A. Вывести оба массива на экран.
- 10) Написать функцию, которая определяет, является ли последовательность целых чисел N знакопеременной. Возвратить 1 – если является, 0 – в противном случае.
- 11) Написать функцию, возвращающую процент отрицательных чисел последовательности из N целых чисел.
- 12) Написать функцию, которая определяет и возвращает сумму натуральных чисел последовательности N, порядковые номера которых являются числами Фибоначчи.

Перечень вопросов к зачёту

Перечень компетенций, проверяемых оценочным средством: ПК-5 – готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ.

- 1) Среда программирования .NET. Её основные компоненты.
- 2) Структура программы на языке C#.
- 3) Создание консольных приложений на языке C#.
- 4) Создание Windows приложений на языке C#.

- 5) Использование переменных в C#. Типы переменных.
- 6) Значимые и ссылочные типы переменных.
- 7) Использование массивов в C#.
- 8) Использование методов в C#.
- 9) Основы объектно-ориентированного программирования.
- 10) Использование объектов в C#.
- 11) Метод Main(). Его значение и использование.
- 12) Условные операторы в C#. Их синтаксис и назначение.
- 13) Оператор выбора в C#. Его синтаксис и назначение.
- 14) Операторы цикла типа for в C#. Их синтаксис и назначение.
- 15) Операторы цикла типа while в C#. Их синтаксис и назначение.
- 16) Потоки ввода-вывода в C#. Их типы и назначение.
- 17) Работа с файлами и каталогами в C#.
- 18) Делегаты в C#. Их синтаксис и назначение.
- 19) События в C#. Их синтаксис и назначение.
- 20) Графические средства в C#.
- 21) Инкапсуляция и наследование объектов. Примеры использования.
- 22) Средства тестирования и отладки в C#.

Компонентом промежуточного контроля по дисциплине «Основы разработки кросс-платформенных приложений» является решение задачи из списка задач по темам к промежуточной аттестации и ответа на теоретический вопрос. Максимальное количество баллов, которые студент может получить за правильное решение задачи, составляет 2 балла. Максимальное количество баллов, которые студент может получить за ответ на контрольный вопрос, составляет 10 баллов.

Рекомендации по оцениванию задачи:

Описание	Баллы
Предоставлен работоспособный программный код, студент может пояснить ход решения, знает назначение команд, может изменить некоторые условия по просьбе преподавателя.	2
Программный код может быть не работоспособен, однако алгоритм решения задачи корректный, студент может пояснить ход решения, знает назначение некоторых команд.	1
Программный код не работает, алгоритм решения не верный, студент не знает назначения отдельных команд.	0

Рекомендации по оцениванию ответа на контрольный вопрос:

Описание	Баллы
Студент владеет теоретическими знаниями по данному разделу, что подтверждается его ответами на дополнительные вопросы; студент умеет правильно объяснять теоретический материал, иллюстрируя его примерами;	8-10
Студент владеет теоретическими знаниями по данному разделу, что подтверждается его ответами на дополнительные вопросы, при ответе студент допускает незначительные ошибки; студент умеет правильно объяснять теоретический материал;	5-7
Теоретический материал не усвоен или усвоен частично, студент не может предоставить чёткий ответ на поставленный вопрос; студент затрудняется привести примеры, поясняющие ответы на вопросы;	0-4

Критерии оценки:

- оценка «**зачтено**»: студент получил не менее 5 баллов за контрольный вопрос, не менее 1 балла за каждое выполненное из 7 заданий, и не менее 1 балла за каждую из 22 индивидуальных задач.

- оценка «**незачёт**»: студент получил менее 5 баллов за контрольный вопрос, или менее 1 балла хотя бы за одно выполненное из 7 заданий, или менее 1 балла хотя бы за одну из 22 индивидуальных задач.

Оценка	
Незачёт	Зачтено
<ul style="list-style-type: none">• студент получил менее 5 баллов за контрольный вопрос;• менее 1 балла хотя бы за одно выполненное из 7 заданий;• менее 1 балла хотя бы за одну из 22 индивидуальных задач	<ul style="list-style-type: none">• студент получил не менее 5 баллов за контрольный вопрос;• не менее 1 балла за каждое выполненное из 7 заданий;• не менее 1 балла за каждую из 22 индивидуальных задач

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

5.1 Основная литература

- 1) Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод. пособие/ Ю.В. Кольцов [и др.]. – Краснодар: Кубанский гос. ун-т, 2015. – 111 с.
- 2) Павловская, Т.А. С#. Программирование на языке высокого уровня: учебник для вузов / Т. А. Павловская. - Санкт-Петербург [и др.]: Питер, 2014. – 432 с.

- 3) Суханов, М.В. Основы Microsoft .NET Framework и языка программирования С# : учебное пособие / М.В. Суханов, И.В. Бачурин, И.С. Майоров ; Министерство образования и науки Российской Федерации, Федеральное государственное автономное образовательное учреждение высшего профессионального образования Северный (Арктический) федеральный университет им. М.В. Ломоносова. - Архангельск : ИД САФУ, 2014. - 97 с. : схем., табл., ил. - Библиогр. в кн. - ISBN 978-5-261-00934-4 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=312313>.

5.2 Дополнительная литература

- 1) Кулямин В.В. Технологии программирования. Компонентный подход: учебное пособие. – М.: Интернет-Университет Информационных Технологий: БИНОМ. Лаборатория знаний, 2010. – 463 с.
- 2) Кариев, Ч.А. Разработка Windows-приложений на основе Visual С# : учебное пособие / Ч.А. Кариев. - Москва : Интернет-Университет Информационных Технологий, 2007. - 768 с. - (Основы информационных технологий). - ISBN 978-5-9556-0080-2 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233307>
- 3) Дунаев, С. Java для Internet в Windows и Linux / С. Дунаев. - Москва : Диалог-МИФИ, 2004. - 490 с. : табл., схем., ил. - ISBN 5-86404-182-2 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=89379>

6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- 1) В.А. Биллиг. Основы программирования на С#. [Электронный ресурс]. – <http://www.intuit.ru/studies/courses/2247/18/info>
- 2) Полное руководство по языку программирования С# 7.0 и платформе .NET 4.7 (<https://metanit.com/sharp/tutorial>)

7. Методические указания для обучающихся по освоению дисциплины

По курсу предусмотрено проведение лабораторных занятий, которые выполняются на компьютере с использованием среды программирования MS Visual Studio (или аналогичной), итогового зачёта.

Важнейшим этапом курса является самостоятельная работа по дисциплине с использованием указанных литературных источников и методических указаний автора курса.

При самостоятельной работе студентов необходимо изучить литературу, приведенную в перечнях выше, для углубления познаний в области разработки приложений .NET Framework на языке С#, для изучения специализированных инструментов и синтаксических конструкций языка. Разрабатывая решение новой задачи, студент должен уметь выбрать эффективные и надёжные структуры данных для представления информации, подобрать соответствующие алгоритмы для их обработки, учесть специфику языка программирования, на котором будет выполнена реализация. Студент должен уметь выполнять тестирование и отладку алгоритмов решения задач с целью обнаружения и устранения в них ошибок.

В качестве систем программирования для решения задач и изучения методов и алгоритмов, приведённых в лекциях, рекомендуется использовать на практических занятиях и при самостоятельной работе среду разработки MS Visual Studio 2013 или новее. Для эффективного программирования рекомендуется использовать встроенные отладчики.

Виды, формы СР, формы контроля приведены выше в данном документе.

Для лучшего освоения дисциплины при защите ЛР студент должен ответить на несколько вопросов.

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

8.1 Перечень информационных технологий

– Проверка лабораторных работ и консультирование посредством электронной почты.

8.2 Перечень необходимого программного обеспечения

– Среда разработки MS Visual Studio

8.3 Перечень информационных справочных систем

- 1) Электронная библиотечная система eLIBRARY.RU (<http://www.elibrary.ru>)
- 2) Справочно-правовая система «Консультант Плюс» (<http://www.consultant.ru>)

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

№	Вид работ	Материально-техническое обеспечение дисциплины и оснащённость
1)	Лабораторные занятия	Лаборатория, укомплектованная специализированными техническими средствами обучения – компьютерный класс, с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета
2)	Групповые (индивидуальные) консультации	Аудитория (кабинет), укомплектованная маркерной доской и оснащенная компьютером.
3)	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная техническими средствами обучения – компьютерами с соответствующим программным обеспечением
4)	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.