

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет математики и компьютерных наук

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование .NET

Кафедра дискретной математики и информатики факультета математики и компьютерных наук

Образовательная программа

02.03.02 - Фундаментальная информатика и информационные технологии

Профиль подготовки
Информатика и компьютерные науки

Уровень высшего образования
бакалавриат

Форма обучения
очная

Статус дисциплины: вариативная по выбору

Махачкала, 2017

Аннотация рабочей программы дисциплины

Дисциплина “Программирование .NET” входит в вариативную по выбору часть образовательной программы бакалавриата по направлению 02.03.02 - Фундаментальная информатика и информационные технологии.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с современными знаниями в области языков и методов программирования: методы программирования на базе языка C# (и среды MS Visual Studio), типы данных и операции, управляющие структуры и визуальные компоненты, создание консольных и графических (на основе форм), объектно-ориентированное и событийно управляемое программирование.

Дисциплина нацелена на формирование следующих компетенций выпускника: общепрофессиональных – ОПК-1; профессиональных – ПК-3, ПК-5.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лабораторные занятия.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: промежуточного контроля в форме 3 контрольных работ и итогового — в форме зачета.

Объем дисциплины составляет 3 зачетных единиц, в том числе в академических часах по видам учебных занятий.

Семестр	Учебные занятия						Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)
	в том числе						
	Контактная работа обучающихся с преподавателем					СРС, в том числе экзамен	
	Все го	из них					
Лекции		Лабораторные занятия	Практические занятия	КСР	Консультации		
5	108	36	36			36	зачет

1. Цели освоения дисциплины

Целью освоения дисциплины (модуля) «Программирование .NET» являются:

- получение фундаментальных знаний по основам современных информационных технологий;

- формирование представлений об основах программирования: знакомство со структурами данных, составление алгоритма решения задачи, реализация алгоритма на языке программирования, отладка и тестирование программы.

Задачей изучения дисциплины является создание и развитие практических навыков по использованию фундаментальных теоретических знаний в области реализации современных и перспективных парадигм программирования

2. Место дисциплины в структуре ООП бакалавриата

Дисциплина «Программирование .NET» входит в вариативную часть образовательной программы бакалавриата по направлению 02.03.02 - Фундаментальная информатика и информационные технологии и является дисциплиной по выбору.

Дисциплина «Программирование .NET» призвана содействовать знакомству студентов с языками и методами программирования и является курсом, для освоения которого необходимы теоретические знания и практические навыки, полученные по дисциплинам «Основы информатики», «Языки и методы программирования», «Дискретная математика».

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения) .

Компетенции	Формулировка компетенции из ФГОС ВО	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)
ОПК-1	способностью использовать базовые знания естественных наук, математики и информатики, основные факты, концепции, принципы теорий, связанных с фундаментальной информатикой и информационными технологиями	Знать: о современных и перспективных информационных технологиях. Уметь: делать обзор и анализ технологии, готовить реферат и представлять современное состояние проблемы. Владеть: навыками подготовки и оформления научной работы: реферата или научного обзора.

ПК-3	способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям	<p>Знать: структуру программы на языке высокого уровня, основные управляющие конструкции, значимые и ссылочные типы</p> <p>Уметь: разрабатывать ясные и надежные алгоритмы для несложных задач, избегая рекурсии, вложенных вызовов, непредсказуемых ситуаций, и составлять адекватные программы с обработкой исключительных ситуаций, с отладкой и достаточно полным тестированием.</p> <p>Владеть: начальными навыками разработки алгоритмов и программ, приемами выбора простых и структурированных типов для представления несложных информационных объектов, принятых считать «стандартными», навыками составления программ (20-50 строк) с использованием методов.</p>
ПК-5	способностью критически переосмысливать накопленный опыт, изменять при необходимости вид и характер своей профессиональной деятельности	<p>Знания: основные понятия, структуры и инструментарий, которые применяются в языках программирования, основные структуры и типы данных, основные методы проектирования и разработки компьютерных программ/</p> <p>Уметь: применять методы программирования при написании программ, определять структуры данных при проектировании алгоритмов, разбивать решение сложных задач на более простые</p> <p>Навыки: освоить язык C# и использовать его при написании программ</p>

4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 3 зачетных единиц, 108 академических часов.

4.2. Структура дисциплины.

№ п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Самостоятельная работа	Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные занятия	Контроль самост. раб.		
Модуль 1. Базовые конструкции языка C#									
1	Введение в платформу .Net	5	1-4	4		4		4	Прием лабораторных работ, реферат
2	Системные типы данных	5	5-6	4		4		4	Прием лабораторных работ

3	Управляющие конструкции языка	5	7-9	4		4		4	Прием лабораторных работ
	Итого по модулю 1:		36	12		12		12	Контрольная работа
	Модуль 2. Работа со строками								
1	Класс String	5	10-13	6		6		6	Прием лабораторных работ
2	Класс StringBuilder	5	14-17	6		6		6	Прием лабораторных работ
	Итого по модулю 2:		36	12		12		12	
	Модуль 3. Построение методов								
1	Понятие перегрузки методов	5	1-4	3		3		3	Прием лабораторных работ
2	Массивы	5	5-8	3		3		3	Прием лабораторных работ
3	Основные принципы ООП	5	9-10	3		3		3	Прием лабораторных работ
4	Наследование и полиморфизм	5	11-17	3		3		3	Прием лабораторных работ
	Итого по модулю 3:		36	12		12		12	Зачет
	ИТОГО:		108	36		36		36	

4.3. Содержание дисциплины, структурированное по темам (разделам).

Модуль 1. Базовые конструкции языка C#

Тема 1. Введение в платформу .Net

Основные принципы построения платформы .Net. Понятия CLR, CTS, CLS.

Сборки, пространства имен и типы. Структура проекта и приложения в среде Visual Studio.

Тема 2. Системные типы данных

Иерархия для типов данных. Внутренние типы. Операции. Преобразования типов.

Тема 3. Управляющие конструкции языка

Операции языка. Условный оператор. Оператор выбора. Операторы цикла.

Модуль 2. Работа со строками

Классы String, StringBuilder. Сравнение строк.

Модуль 3. Построение методов

Тема 1. Понятие перегрузки методов

Методы и модификаторы параметров. Перегрузка методов. Перегрузка операций.

Тема 2. Массивы

Класс Array. Ступенчатые массивы

Тема 3. Основные принципы ООП

Тип класса C#. Понятия инкапсуляции, наследования и полиморфизма

Тема 4. Наследование и полиморфизм

Родительский класс System.Object. Типы наследования. Переопределение методов: virtual и override. Интерфейсы.

Темы лабораторных занятий

Модуль 1. Базовые конструкции языка C#

Тема 1. Введение в платформу .Net

Основные принципы построения платформы .Net. Понятия CLR, CTS, CLS.

Сборки, пространства имен и типы. Структура проекта и приложения в среде Visual Studio.

Тема 2. Системные типы данных

Иерархия для типов данных. Внутренние типы. Операции. Преобразования типов.

Тема 3. Управляющие конструкции языка

Операции языка. Условный оператор. Оператор выбора. Операторы цикла.

Модуль 2. Работа со строками

Классы String, StringBuilder. Сравнение строк.

Модуль 3. Построение методов

Тема 1. Понятие перегрузки методов

Методы и модификаторы параметров. Перегрузка методов. Перегрузка операций.

Тема 2. Массивы

Класс Array. Ступенчатые массивы

Тема 3. Основные принципы ООП

Тип класса C#. Понятия инкапсуляции, наследования и полиморфизма

Тема 4. Наследование и полиморфизм

Родительский класс System.Object. Типы наследования. Переопределение методов: virtual и override. Интерфейсы.

5. Образовательные технологии

В процессе преподавания дисциплины «Практикум на ЭВМ» применяются следующие образовательные технологии: лекционно-зачетная система обучения, технология развития критического мышления, изложение теоретических положений в ходе лекционных занятий с применением современного интерактивного презентационного оборудования, проведение групповых дискуссий. Удельный вес занятий, проводимых в интерактивных формах, составляет не менее 70% аудиторных занятий. Основная литература, презентации и конспекты лекций предоставляются студентам в электронном виде.

6. Учебно-методическое обеспечение самостоятельной работы студентов.

6.1 Виды самостоятельной работы и её контроля

Самостоятельная работа студентов складывается из:

- проработки лекционного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных на лекциях упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к сдаче промежуточных форм контроля (контрольных работ).

Пакет заданий для самостоятельной работы выдается по истечению месяца с начала семестра, определяются предельные сроки их выполнения и сдачи.

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
1.	Проработка лекционного материала	Контрольный фронтальный опрос	См. разделы 7.3, 8, 9 данного документа
2.	Изучение рекомендованной литературы и материалов соответствующих форумов интернет	Контрольный фронтальный опрос, прием и представление рефератов.	См. разделы 7.3, 8, 9 данного документа
3.	Подготовка к отчетам по лабораторным работам.	Проверка выполнения работ, опрос по теме работы.	См. разделы 7.3, 8, 9 данного документа
4.	Подготовка к сдаче промежуточных форм контроля	Контрольные работы по каждому модулю.	См. разделы 7.3, 8, 9 данного документа

Типовое задание для самостоятельной работы

Нажатием на кнопку организовать диалог по выбору файла любых форматов, поддерживаемых WindowsMediaPlayer, и проиграть его содержимое в окне.

Указания к решению. Поместим на форму кнопку и невидимый объект openFileDialog1, затем для размещения элемента управления

WindowsMediaPlayer на форму внесем предварительно его в раздел General панели инструментов: вызовем контекстное меню раздела General панели инструментов, выберем пункт "Choose Items", в открывшемся окне для выбора компонент на вкладке COM Components поставим галочку рядом с Windows Media Player и нажмем ОК. Появившийся в разделе General появится элемент Windows Media Player перетащим на форму (или дважды щелкнем по этому элементу) и растянем его по ширине формы. Во вкладке Properties изменим имя полученного объекта на «WMP1».

В обработчике щелчка по кнопке (подпишемся на щелчок по кнопке) занесем имя выбранного файла в свойство WMP1.URL.

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1 Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.

Перечень компетенций с указанием этапов их формирования приведен в описании образовательной программы.

Компетенция	Знания, умения, навыки	Процедура освоения
ОПК-1	<p>Знать: о современных и перспективных информационных технологиях.</p> <p>Уметь: делать обзор и анализ технологии, готовить реферат и представлять современное состояние проблемы.</p> <p>Владеть: навыками подготовки и оформления научной работы: реферата или научного обзора.</p>	<p>Устный опрос, письменный опрос</p>
ПК-3	<p>Знать: структуру программы на языке высокого уровня, основные управляющие конструкции, значимые и ссылочные типы</p> <p>Уметь: разрабатывать ясные и надежные алгоритмы для несложных задач, избегая рекурсии, вложенных вызовов, непредсказуемых ситуаций, и составлять адекватные программы с обработкой исключительных ситуаций, с отладкой и достаточно полным тестированием.</p> <p>Владеть: начальными навыками разработки алгоритмов и программ, приемами выбора простых и структурированных типов для представления несложных информационных объектов, принятых считать «стандартными», навыками составления программ (20-50 строк) с использованием методов.</p>	<p>Проработка конспектов лекций и подготовка к контрольным работам. Подготовка и презентация реферата.</p>
ПК-5	<p>Знания: основные понятия, структуры и инструментарий, которые применяются в языках программирования, основные структуры и типы данных, основные методы проектирования и разработки компьютерных программ/</p> <p>Уметь: применять методы программирования при написании программ, определять структуры данных при проектировании алгоритмов, разбивать решение сложных задач на более простые</p> <p>Навыки: освоить язык C# и использовать его при написании программ</p>	<p>Проработка конспектов лекций и подготовка к контрольным работам. Подготовка и презентация реферата.</p>

7.2 Описание показателей и критериев оценивания компетенций, описание шкал оценивания.

ОПК-1

Схема оценки уровня формирования компетенции «способность использовать базовые знания естественных наук, математики и информатики, основные факты, концепции, принципы теорий, связанных с фундаментальной информатикой и информационными технологиями»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый		Знать языки программирования.	Уметь разрабатывать ясные и надежные программы.	Уметь разрабатывать программы в Visual Studio

ПК-3

Схема оценки уровня формирования компетенции «способность использовать современные инструментальные и вычислительные средства».

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Выполнение контрольных работ без ошибок или с несущественными недочетами.	Выполнение базовой части контрольной работы (50 % заданий) без существенных недочетов. Составление реферата без презентации.	Выполнение 60-70% контрольной работы без существенных недочетов. Представление реферата в аудитории с обсуждением.	Выполнение более 70% заданий контрольной работы без существенных недочетов. Составление реферата и его успешная презентация с ответами на вопросы в аудитории.

ПК-5

Схема оценки уровня формирования компетенции «способностью критически переосмысливать накопленный опыт, изменять при необходимости вид и характер своей профессиональной деятельности».

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично

Пороговый		Выполнение базовой части контрольной работы (50 % заданий) без существенных недочетов. Составление реферата без презентации.	Выполнение 60-70% контрольной работы без существенных недочетов. Представление реферата в аудитории с обсуждением.	Выполнение более 70% заданий контрольной работы без существенных недочетов. Составление реферата и его успешная презентация с ответами на вопросы в аудитории.
-----------	--	--	--	--

7.3. Типовые контрольные задания

7.3.1. Темы рефератов:

1. Консольные приложения Delphi и C#. Сравнительная характеристика.
2. Компьютерный вывод рекуррентных формул (на примере задачи разбиения).
3. Длинная целочисленная арифметика.
4. Числа Фибоначчи. Вычислительные аспекты.
5. Настольные графические приложения.
6. Проекты с несколькими формами.
7. Компьютерная графика в C#-проектах.
8. OpenGL в проектах C#.
9. Проекты с применением веб-камер. Распознавание цифр.
10. Визуальные компоненты Visual C#.
11. Класс BigInteger в задачах криптографии

7.3.2. Примерные упражнения и задания к лабораторным занятиям и для самопроверки

Задание 1. Найти время вычисления операций +, -, *, /.

Указания к решению. Во-первых, необходим цикл. Во-вторых, пустой цикл или цикл с присвоением, чтобы время этого цикла (время на организацию цикла) отнять. Самое главное – нужно учитывать мультизадачность, что совсем трудно; то, что при разных запусках наша программа запускается в одном потоке – утешение малое, т.к. этому потоку будет отводиться малое время.

Целесообразно программу запускать 10 раз (программным путем).

Задание 2. Составить программу с двумя кнопками «Действия с блокнотом» и «Запуск калькулятора» и текстовым контейнером.

При щелчке на первую кнопку запустить программу Блокнот, ввести в первую строку несколько нулей, во вторую - несколько единиц, затем сохранить текст в текущем каталоге в файле 1234.txt (в предположении отсутствия одноименного файла), после чего программа закроет Блокнот.

При щелчке по второй кнопке проверить, запущено ли стандартное приложение Калькулятор, в случае, если не запущено, программа должна запустить его, затем передачей в него кодов нажатых клавиш выполнить эмуляцию видимых действий по суммированию $0+1+2+ \dots + 100$.

При щелчке по контейнеру текста выполнить бесконечный вывод в него буквы «А».

7.3.3. Примерный перечень заданий для текущего контроля

Упр. 1. Введите с клавиатуры n , объявите одномерный массив a из n целых элементов и выведите построчно на экран: а) с использованием индексов, б) с помощью оператора `foreach`.

Упражнение 2. Заполнить прямоугольный массив $n*m$ ($n=m=4$) случайными целыми числами $< k=20$ и просуммировать все четные элементы, сумму вывести в формате с 5 знаками.

Упражнение 3. Измените решение первого упражнения на вывод отсортированного массива, добавить вывод индекса значения 0 в отсортированном массиве.

Упражнение 4 (см. в тексте лекции). Обратите внимание на аргумент функции `Main`. Аргумент `string [] args` является массивом строк. Это параметры, принимаемые из командной строки (задаются после имени программы с пробельным разделителем). Перейдем в Windows в режим командной строки, используя главное меню. Затем, используя команды `cd`, перейдем в папку, которая содержит исполняемый файл следующей программы. Пусть это `A.exe`. Наберите командной строке `A 12 13 14` и нажмите `Enter` для выполнения.

Упражнение 5. Программа должна удалить из папки `C:\SomeDir` своего компьютера все файлы с расширением `bmp`.

Упражнение 6. Введите предложение и подсчитайте количество слов в нем, учитывая всевозможные разделители.

Упражнение 7. Заданную с клавиатуры строку выведите в верхнем и нижнем регистре.

Упражнение 8. Создание коллекции - строки статуса с панелями; в первую панель выводятся текущие координаты курсора мыши, во вторую – текущее время. Для обеих панелей предусмотреть всплывающие подсказки («координаты мыши» и текущую дату соответственно).

Упражнение 9. При запуске программы на поверхности формы нарисовать $m=10$ рядов квадратных кнопок по $m=10$ кнопок в ряду, размеры кнопки – $30*30$. Затем выполнить дрожание кнопок в 10 итерациях.

Упражнение 10. Создать проект из двух форм. При нажатии на кнопку первой формы предьявляется вторая форма, а первая начинает «прогулку» по невидимой окружности с постепенным усилением прозрачности.

Вопросы к зачету

1. Основные принципы построения платформы .Net. Понятия CLR, CTS, CLS.
2. Сборки, пространства имен и типы. Структура проекта и приложения в среде Visual Studio.
3. Иерархия для типов данных.
4. Внутренние типы.
5. Преобразования типов.
6. Операции языка.
7. Условный оператор.
8. Оператор выбора.
9. Операторы цикла.
10. Классы `String`, `StringBuilder`. Сравнение строк.
11. Методы и модификаторы параметров.
12. Перегрузка методов.
13. Перегрузка операций.

14. Класс Array.
15. Ступенчатые массивы
16. Тип класса C#.
17. Понятия инкапсуляции, наследования и полиморфизма
18. Родительский класс System.Object.
19. Типы наследования.
20. Переопределение методов: virtual и override.
21. Интерфейсы.

7.4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Общий результат выводится как интегральная оценка, складывающаяся из текущего контроля - 50% и промежуточного контроля - 50%. Текущий контроль по дисциплине включает:

- посещение занятий - 30 баллов,
- выполнение лабораторных заданий – 20 баллов,
- выполнение домашних (аудиторных) контрольных работ - 50 баллов. Промежуточный контроль по дисциплине включает:
 - устный опрос - 30 баллов,
 - письменная контрольная работа - 50 баллов,
 - тестирование - 20 баллов.

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

а) основная литература:

1. Эндрю Троелсен. Язык программирования C# 5.0 и платформа .NET 4.5 (6е издание). Издательство: Вильямс, 2013.
2. Герберт Шилдт. C# 4.0. Полное руководство. Издательство: Вильямс, 2015.
3. Джозеф Албахари, Бен Албахари. C# 5.0. Справочник. Полное описание языка (5-е издание). Издательство: Вильямс, 2013. - 1054 стр.
- Культин Н.Б. Microsoft C# в задачах и примерах. – СПб.: БХВ-Петербург, 2009. – 320 с.

Дополнительная:

4. Лабор В.В. Си Шарп: Создание приложений для Windows. – Мн.: Харвест, 2011. – 384 с.
5. Алексеев В.Е., Таланов В.А. Графы и алгоритмы. Структуры данных. Модели вычислений. - М.: Интернет-университет информационных технологий; БИНОМ. Лаборатория знаний, 2006.
6. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. - М.: Вильямс, 2006.

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

<http://citforum.ru/>

<http://www.intuit.ru/>

9. Методические указания для обучающихся по освоению дисциплины.

При решении лабораторных заданий программистский подход непременно должен присутствовать (без него решение не будет полноценным), однако, он не должен заслонять сугубо математические (доказательство и др.) и алгоритмические (построение, оптимизация, верификация и др.) аспекты.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение: Microsoft Visual Studio Express, Microsoft Windows, Ubuntu Linux, Skype.

Также студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.

Имеется необходимая литература в библиотеке, медиапроектор и компьютер для проведения лекций-презентаций.

Лабораторные занятия проводятся в компьютерных классах с необходимым программным обеспечением. Вся основная литература предоставляется студенту в электронном формате.