

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

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование в .NET

Кафедра дискретной математики и информатики
факультета математики и компьютерных наук

Образовательная программа бакалавриата

02.03.02 Фундаментальная информатика и информационные технологии

Направленность (профиль) программы:
Информатика и компьютерные науки

Форма обучения:

очная

Статус дисциплины: входит в часть,
формируемую участниками образовательных отношений

Махачкала, 2022

Рабочая программа дисциплины «Программирование в .NET» составлена в 2022 году в соответствии с требованиями ФГОС ВО - бакалавриат по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии

от «23» августа 2017 г. № 808.

Разработчик(и): кафедра дискретной математики и информатики, Раджабова Наима Шамильевна, к.ф.-м.н., доцент.

Рабочая программа дисциплины одобрена:

на заседании кафедры дискретной математики и информатики от «28» февраля 2022 г., протокол № 6;

зав. кафедрой  Магомедов А. М.
(подпись)

и

на заседании Методической комиссии факультета математики и компьютерных наук от «24» марта 2022 г., протокол № 4;

председатель  Ризаев М. К.
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением «31» марта 2022 г.

Начальник УМУ  Гасангаджиева А. Г.
(подпись)

1. Цели освоения дисциплины

Целью освоения дисциплины (модуля) «Программирование в .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 ОПОП бакалавриата

Дисциплина «Программирование в .NET» входит в часть, формируемую участниками образовательных отношений ОПОП бакалавриата по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии.

Дисциплина «Программирование в .NET» призвана содействовать знакомству студентов с языками и методами программирования для проектирования сложных программных систем. Для освоения дисциплины необходимы теоретические знания и практические навыки, полученные по дисциплинам «Основы информатики», «Языки и методы программирования», «Дискретная математика». Также необходимо знание основ одного из объектно-ориентированных языков программирования, умение проектировать классы, приобретенные на занятиях по «Языкам и методам программирования» и Java-программированию.

Результаты освоения данной дисциплины будут востребованы на занятиях по «Распределенным объектным технологиям», «Современным веб-технологиям» и при подготовке выпускной квалификационной работы.

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

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций (в соответствии ОПОП)	Планируемые результаты обучения	Процедура освоения

<p>ОПК-3. Способен к разработке алгоритмических программных решений в области системного и прикладного программирования, математических, информационных, имитационных 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>	<p>ОПК-3.1. Знает методы теории алгоритмов, методы системного и прикладного программирования, основные положения и концепции в области математических, информационных и имитационных моделей.</p>	<p>Знает: принципы объектно-ориентированного моделирования и разработки на основе международного стандарта RUP. Умеет: создавать различные типы диаграмм, используя CASE-инструменты IDE и Visual Paradigm. Владеет: навыками создания диаграмм вариантов использования и диаграмм классов.</p>	<p>Конспектирование лекций, подготовка и защита реферата, участие в дискуссиях. Разработка и реализация проектов на лабораторных занятиях.</p>
	<p>ОПК-3.2. Умеет соотносить знания в области программирования, интерпретацию прочитанного, определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем.</p>	<p>Знает: принципы объектно-ориентированного моделирования и разработки на основе международного стандарта RUP. Умеет: создавать полную модель прецедентов. Владеет: навыками создания диаграмм вариантов использования и диаграмм классов.</p>	

	<p>ОПК-3.3. Имеет практический опыт применения разработки программного обеспечения.</p>	<p>Знает: графическую нотацию языка UML и классы метамодели языка UML.</p> <p>Умеет: отображать графическую нотацию языка UML в программы на языках программирования Delphi, C#, Java, C++ (по выбору студента), а также представлять программы на языках программирования Delphi, C#, Java, C++ (по выбору студента) с помощью графической нотации UML.</p> <p>Владеет: навыками использования языка UML с помощью CASE-инструментов при проектировании программного обеспечения.</p>	
--	--	---	--

<p>ПК-4. Способность применять в профессиональной деятельности современные языки программирования и методы параллельной обработки данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии.</p>	<p>ПК-4.1. Знает современные языки программирования и методы параллельной обработки данных. Знаком с содержанием Единого Реестра Российских программ для электронных вычислительных машин и баз данных.</p>	<p>Знает: основные понятия, структуры и инструментарий, которые применяются в языках программирования, основные структуры и типы данных, основные методы проектирования и разработки компьютерных программ. Умеет: применять методы программирования при написании программ, определять структуры данных при проектировании алгоритмов, разбивать решение сложных задач на более простые Владеет: навыками использования языка C# при написании программ</p>	<p>Проработка конспектов лекций и подготовка к контрольным работам. Подготовка и презентация реферата. Выполнение лабораторных работ</p>
	<p>ПК-4.2. Умеет реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии.</p>	<p>Знает: структуру программы на языке высокого уровня, основные управляющие конструкции, значимые и ссылочные типы Умеет: разрабатывать ясные и надежные алгоритмы для несложных задач, избегая рекурсии, вложенных вызовов, непредсказуемых ситуаций, и составлять адекватные программы с обработкой исключительных</p>	

		<p>ситуаций, с отладкой и достаточно полным тестированием.</p> <p>Владеет: 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>ПК-4.3. Имеет практический опыт разработки интеграции информационных систем.</p>	<p>Знает: международные и российские стандарты жизненного цикла ПО.</p> <p>Умеет: разрабатывать техническое задание для проекта по заданному шаблону.</p> <p>Владеет: навыками объектно-ориентированного проектирования задач в различных предметных областях.</p>	

4. Объем, структура и содержание дисциплины.

4.1 Объем дисциплины составляет 3 зачетные единицы, 108 академических часов.

4.2 Структура дисциплины.

4.2.1. Структура дисциплины в очной форме

п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа	Контроль сам. работы	
Модуль 1. Базовые конструкции языка C#									
1	Введение в платформу .Net	5	1-2	4		4	4		Прием лабораторных работ, реферат
2	Системные типы данных	5	3-4	4		4	4		Прием лабораторных работ, реферат
3	Управляющие конструкции языка	5	5-6	4		4	4		Прием лабораторных работ
Итого по модулю 1:			36	12		12	12		Контрольная работа
Модуль 2. Работа со строками									
4	Класс String	5	7-9	6		6	10		Прием лабораторных работ.
5.	Класс StringBuilder	5	10-12	2		2	10		
Итого по модулю 2			36	8		8	20		Контрольная работа
Модуль 3. Построение методов									
6.	Понятие перегрузки методов	5	13	2		2	4		Прием лабораторных работ
7.	Массивы	5	14	2		2	4		Прием лабораторных работ
8.	Основные принципы ООП	5	15	2		2	2		Прием лабораторных работ, реферат
9.	Наследование и полиморфизм	5	16	2		2	2		Прием лабораторных работ.

10.	Событийно-управляемое программирование	5	17-18	2		2	4		Прием лабораторных работ.
	Итого по модулю 3		36	10		10	16		Контрольная работа
	ИТОГО:		108	36		36	48		Зачет

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

4.3.1. Содержание лекционных занятий по дисциплине.

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

Тема 1. Класс String

Тема 2. Класс 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.

Тема 5. Событийно-управляемое программирование.

Проектирование типа, поддерживающего событие. Реализация события.

4.3.2 Содержание лабораторных занятий по дисциплине

Модуль 1. Базовые конструкции языка C#

Лабораторная работа 1. Сборки, пространства имен и типы. Структура проекта и приложения в среде VisualStudio.

Лабораторная работа 2. Иерархия для типов данных. Внутренние типы. Операции. Преобразования типов.

Лабораторная работа 3. Операции языка. Условный оператор. Оператор выбора. Операторы цикла.

Модуль 2. Работа со строками

Лабораторная работа 4. Классы String, StringBuilder. Сравнение строк.

Модуль 3. Построение методов

Лабораторная работа 5. Перегрузка методов и операций

Лабораторная работа 6. Класс Array. Ступенчатые массивы

Лабораторная работа 7. Конструирование класса C#.

Лабораторная работа 8. Иерархия типов C#. Переопределение методов: virtual и override. Интерфейсы.

Лабораторная работа 9. Создание приложений Калькулятор, Блокнот.

Лабораторная работа 10. Создание приложения Анализатор текста.

Лабораторная работа 11. Обработка списков.

5. Образовательные технологии

В процессе преподавания дисциплины «Программирование в .NET» 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. Типовые контрольные задания

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

7.1.1 Темы рефератов:

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

7.1.2 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».

7.1.3 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.

7.1.4 Контрольные вопросы

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.

7.2 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) адрес сайта курса

<https://explorecoursesit.blogspot.com/>

б) 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.
4. Рихтер Дж. CLR via C# Программирование на платформе Microsoft .NET Framework 4.0 на языке C#.Мастер-класс.СПб.: Питер, 2015.
5. Павловская Т.А. Программирование на языке высокого уровня C# [Электронный ресурс] / Т.А. Павловская. — 2-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 245 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/73713.html>

б) дополнительная:

6. Токманцев Т.Б. Алгоритмические языки и программирование: учебное пособие / Токманцев Т.Б.— Е.: Уральский федеральный университет, ЭБС АСВ, 2013. 104— с.
7. Мейер Б. Инструменты, алгоритмы и структуры данных [Электронный ресурс] / Б. Мейер. — 2-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 542 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/73680.html>

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

Видеокурсы лекций:

- 1) <https://www.coursera.org/>
- 2) <https://www.udacity.com/>
- 3) <https://www.intuit.ru/>

Форумы по компьютерным наукам и программированию:

- 1) www.stackoverflow.com
- 2) <http://www.cyberforum.ru/>
- 3) <http://citforum.ru/http://www.intuit.ru/>

10. 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.

11. 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.

12. 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.