

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

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование в .NET

Кафедра дискретной математики и информатики
факультета математики и компьютерных наук

Образовательная программа
02.04.02 Фундаментальная информатика и информационные технологии

Профиль подготовки
Информатика и компьютерные науки

Уровень высшего образования
магистратура

Форма обучения
Очно-заочная

Статус дисциплины: входит в часть ОПОП,
формируемую участниками образовательных отношений

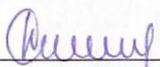
Рабочая программа дисциплины «Программирование в .Net» составлена в 2021 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.04.02 Фундаментальная информатика и информационные технологии (уровень магистратуры) от «27» августа 2017 г. № 811.

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

Рабочая программа дисциплины одобрена:
на заседании кафедры информатики от «30» 05 2021 г., протокол № 9.

Зав. кафедрой  Магомедов А. М.
(подпись)

и
на заседании Методической комиссии ФМиКН
от «23» июня 2021 г., протокол № 6.

Председатель  В.Д. Бейбалаев
(подпись)

Рабочая программа дисциплины согласована
с учебно-методическим управлением «09» 07 2021 г.

(Начальник УМУ  Гасангаджиева А.Г.
(подпись)

Аннотация рабочей программы дисциплины

Дисциплина “Программирование в .NET” входит в часть, формируемую участниками образовательных отношений – модуль профильной направленности (Б1.В.01) магистратуры по направлению 02.04.02 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 С# (платформы .NET 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: профессиональных – ПК-1, ПК-4.

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

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: в форме 4 коллоквиумов и промежуточный контроль в форме экзамена. Объем дисциплины составляет 5 зачетных единиц, в том числе в академических часах по видам учебных занятий.

Очно-заочная форма обучения

Семестр	Учебные занятия						Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)
	в том числе						
	Контактная работа обучающихся с преподавателем					СРС, в том числе экзамен	
	Всего	из них					
Лекции		Лабораторные занятия	Практические занятия	КСР	Консультации		
5	180	12	14				118+36 экзамен

1. Цели освоения дисциплины

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

- получение фундаментальных знаний по основам платформы .NET;

- формирование представлений об основах программирования на C#: 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 магистратуры

Дисциплина «Программирование в .NET» входит в часть, формируемую участниками образовательных отношений образовательной программы магистратуры по направлению 02.04.02 Фундаментальная информатика и информационные технологии.

Дисциплина «Программирование в .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». 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 и процедура освоения).

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций (в соответствии с ОПОП)	Планируемые результаты обучения	Процедура освоения
ПК-1 Способность понимать и применять в научно-исследовательской и прикладной деятельности современный математический аппарат, основные законы	. ПК-1.1. Знает основы ведения научной дискуссии и формы устного научного высказывания. ПК-1.2. Умеет вести корректную дискуссию в области информационных технологий, задавать	Знает: основные парадигмы, платформы, методы проектирования и разработки компьютерных программ. Умеет: применять методы программирования при написании программ, определять структуры	Посещение лекций и выполнение лабораторных работ. Проработка материалов лекций, участие в дискуссиях на лекциях. Сдача отчета и коллоквиума.

<p>естествознания, современные языки программирования и программное обеспечение; операционные системы и сетевые технологии.</p>	<p>вопросы и отвечать на поставленные вопросы по теме научной работы. ПК-1.3. Имеет практический опыт владения существующими методами и алгоритмами решения задач цифровой обработки сигналов, использования сети Интернет, аннотирования, реферирования, библиографического разыскания и описания, опыт работы с научными источниками</p>	<p>данных при проектировании алгоритмов, разбивать решение сложных задач на более простые Владеет: навыками программирования на языке C# при написании программ</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. ПК-4.2. Умеет реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии. ПК-4.3. Имеет практический опыт разработки интеграции информационных систем.</p>	<p>Знает: структуру программы на языке высокого уровня, основные управляющие конструкции, значимые и ссылочные типы Умеет: разрабатывать ясные и надежные алгоритмы для стандартных задач, составлять локальные и распределенные программы с обработкой исключительных ситуаций, с отладкой и достаточно полным тестированием. Владеет: навыками разработки алгоритмов л</p>	<p>Проработка материалов лекций и выполнение лабораторных работ. Сдача отчетов и коллоквиума</p>

--	--	--	--

4. Объем, структура и содержание дисциплины.

4.1 Объем дисциплины составляет 5 зачетных единицы, 180 академических часов.

4.2 Структура дисциплины в очно-заочной форме

п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов (в часах)					Формы текущего контроля успеваемости и промежуточной аттестации
				Лекции	Практические занятия	Лабораторные занятия	Самостоятельная работа в т.ч. экзамен		
Модуль 1. Введение в платформу .NET									
1	Основы платформы .Net	1	1	2		2	12	Прием лабораторных работ	
2	Системные типы данных	1	2	2		2	16	Прием лабораторных работ	
Итого по модулю 1:			36	4		4	28	Коллоквиум	
Модуль 2. Управляющие конструкции языка C#									

3.	Операторы языка С#			2		2	14	Прием лабораторных работ	
4.	Массивы и оператор foreach в С#.			2		2	14	Прием лабораторных работ	
	Итого по модулю 2:			4		4	28	Коллоквиум	
Модуль 3. Основы объектно-ориентированного проектирования									
5.	Основные принципы ООП. Принципы SOLID			2		2	16	Прием лабораторных работ	
6.	Построение методов					2	14	Прием лабораторных работ	
	Итого по модулю 3:		36	2		4	30	Коллоквиум	
Модуль 4. Создание веб-API с помощью ASP.NET Core									
7.	Основы веб-сервисов: WCF и REST			2			16	Прием лабораторных работ	
8.	REST в ASP.Net Core					2	16	Прием лабораторных работ	
	Итого по модулю 4:		36	2		2	32	Коллоквиум	
	ИТОГО:		180	12		14	118	Экзамен	

4.3 Содержание дисциплины, структурированное по темам (разделам)

Модуль 1. Базовые конструкции языка С#

Тема 1. Введение в платформу .Net

Основные принципы построения платформы .NET. Понятия CLI, CLR, CTS, IL. Библиотека классов .NET. Сборки, пространства имен и типы. Сценарии использования .NET.

Тема 2. Системные типы данных

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

Модуль 2. Управляющие конструкции языка С#

Тема 3. Операторы языка С#.

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

Тема 4. Массивы и оператор foreach в С#.

Хранение и перебор последовательностей данных с помощью массивов и оператора foreach в C#. Массивы и строки. Класс Array. Ступенчатые массивы. Оператор перебора foreach. Класс String. Класс StringBuilder. Сравнение строк.

Модуль 3. Основы объектно-ориентированного проектирования

Тема 5. Основные принципы ООП. Принципы SOLID

Тип класса C#. Понятия инкапсуляции, наследования и полиморфизма. Основы проектирования классов. Понятие перегрузки методов. Родительский класс System.Object. Типы наследования. Переопределение методов: virtual и override. Интерфейсы. Проектирование типа, поддерживающего событие. Реализация события.

Тема 6. Построение методов

Методы и модификаторы параметров. Перегрузка методов. Перегрузка операций.

Модуль 4. Создание веб-API с помощью ASP.NET Core

Тема 7. Основы веб-сервисов: WCF и REST

История веб-сервисов и сервис-ориентированного подхода. Определение и составляющие сервис-ориентированной архитектуры. Слабо- и сильносвязанные программные системы.

Основные принципы сервис-ориентированного подхода. Основные стандарты веб-сервисов: WSDL, SOAP, WS-Security, WS-Addressing, WSRF. Основные принципы WCF.

Шесть ограничений REST. Методы протокола HTTP. Форматы HTML, XML и JSON.

Тема 8. REST в ASP.Net Core

Создание проекта веб-API с помощью ASP.NET Core. Создание базы данных в памяти для сохранения продуктов. Добавление поддержки операций CRUD.

Тестирование методов действий веб-API из командной оболочки.

4.3.2 Содержание лабораторно-практических занятий по дисциплине

Модуль 1. Введение в платформу .NET

Лабораторная работа 1. Создание приложений .NET на C#.

Инициализация проектов .NET

Добавление и удаление пакетов для проекта .NET

Управление зависимостями пакетов и их обновлениями.

Лабораторная работа 2. Отладка приложений .NET в интерактивном режиме с помощью отладчика Visual Studio Code

Использование отладчика Visual Studio Code с программой .NET.

Создание точек останова и пошаговое выполнение кода для поиска проблем.

Проверка состояния программы на любом шаге выполнения.

Модуль 2. Управляющие конструкции языка C#

Лабораторная работа 3. Операции языка. Условный оператор. Оператор выбора. Операторы цикла. Работа с файлами и каталогами в приложении .NET.

Лабораторная работа 4. Работа с массивами

Класс Array. Ступенчатые массивы Классы String, StringBuilder. Сравнение строк.

Модуль 3. Основы объектно-ориентированного проектирования

Лабораторная работа 5. Вызов методов из библиотеки классов .NET с помощью C#

Перегрузка методов и операций.

Лабораторная работа 6. Конструирование класса C#.

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

Модуль 4. Создание веб-API с помощью ASP.NET Core

Лабораторная работа 7. Создание проекта веб-API с помощью ASP.NET Core.

Создание базы данных в памяти для сохранения продуктов.

Добавление поддержки операций CRUD.

Тестирование методов действий веб-API из командной оболочки.

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

Типовые задания для самостоятельной работы можно найти на сайте

<https://docs.microsoft.com/ru-ru/learn/paths/build-dotnet-applications-csharp/>

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

7.1. Типовые контрольные задания

7.1.1 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».

7.2.3 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 Вопросы к экзамену

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.
22. Основные принципы WCF.
23. Шесть ограничений REST. Методы протокола HTTP. Форматы HTML, XML и JSON.
24. Создание проекта веб-API с помощью ASP.NET Core.

7.3 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 и активность на занятиях - 20 баллов,
- выполнение лабораторных заданий – 30 баллов,
- устный опрос – коллоквиум – 50 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.

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> (дата обращения: 12.04.2018)
 - 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> (дата обращения: 20.04.2018)

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/>
- 4) <https://docs.microsoft.com/ru-ru/learn/paths/build-dotnet-applications-csharp/>

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.

1. 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 Code, Microsoft Windows, Ubuntu Linux, Skype.

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

2. 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.