

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования

«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет математики и компьютерных наук

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Языки программирования

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: входит в обязательную часть ОПОП

Махачкала, 2021

Рабочая программа дисциплины «Языки программирования» составлена в 2021 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.03.02 - Фундаментальная информатика и информационные технологии.

Приказ №808 Минобрнауки России от 23 августа 2017 г.

Разработчик: проф. по специальности 01.01.09 - «Дискретная математика и математическая кибернетика», докт. физ.-мат. наук. Магомедов Абдулкарим Магомедович

Рабочая программа дисциплины одобрена:

на заседании кафедры дискретной математики и информатики от «30» мая 2021 г., протокол № 9.

Зав. кафедрой ММ Магомедов А.М.
(подпись)

и

на заседании Методической комиссии ФМиКН от
«23» июня 2021г., 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 “Языки программирования” входит в обязательную 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: методы программирования на базе языка С# (и среды MS Visual Studio), типы данных и операции, управляющие структуры и визуальные компоненты, создание консольных и графических (на основе форм) приложений, объектно-ориентированное и событийно управляемое программирование.

Дисциплина способствует формированию следующих компетенций выпускника: ОПК-3, ПК-4.

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

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: в форме 2-х коллоквиумов (модулей) и итогового экзамена в конце семестра.

Объем дисциплины в очной форме – 3 зачетные единицы, в том числе в академических часах по видам учебных занятий:

Семестр	Учебные занятия						Контр оль	СРС	Форма промежуточ ой аттестации (зачет, дифференцир ованный зачет), экзамен
	в том числе								
	всего	Контактная работа обучающихся с преподавателем							
		всего	из них						
	Лек ции	Лаборатор ные занятия	Прак. занят ия						
1	108	52	26	26	0		36	20	экзамен

1. Цели освоения дисциплины

Изучение среды MS Visual Studio и языка программирования C# для овладения знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования. Формирование у студентов научного, творческого подхода к освоению технологий, методов и средств производства программного обеспечения.

Основные задачи дисциплины: овладение методами структурного и объектно-ориентированного программирования; обучение разработке алгоритмов; закрепление навыков алгоритмизации и программирования, полученных в 1-м семестре; создание практической базы для качественного прохождения учебной практики и изучения других учебных дисциплин ("Компьютерная графика" и др.).

Ожидаемые результаты:

- усвоение базовых типов и операций языка C#;
- свободное применение управляющих структур;
- умение осознанного выбора типа приложения (консольный, графический) в соответствии с типом задачи;
- ознакомление со структурированными данными (массивы, списки, кортежи и др.) и умение их использовать в программе;
- грамотное использование практически всех визуальных компонент, умение создавать приложение с одной и несколькими формами;
- изучение и использование новых средств компьютерной графики;
- создание функций и обработчиков событий;
- овладение нестандартными средствами языка: длинная арифметика, передача кодов нажатых клавиш, запуск программ, действия с веб-страницами.
- ознакомление с понятием цифровой подписи на уровне активного ее использования.

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина входит в обязательную часть образовательной программы бакалавриата по направлению 02.03.02 - Фундаментальная информатика и информационные технологии и изучается в соответствии с графиком учебного процесса во втором семестре. Изучение предмета производится в течение одного семестра и заканчивается экзаменом.

Дисциплина опирается на знания, полученные в 1 семестре в процессе изучения Основ программирования. В свою очередь, на материал данной дисциплины опираются дисциплины «Компьютерная графика» и «Прикладная теория графов»; знания, умения и навыки, полученные при изучении данной дисциплины, используются, закрепляются и развиваются при проведении учебной практики (после 2 семестра).

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

Код и наименование компетенции из ФГОС ВО	Код и наименование индикатора достижения компетенций	Планируемые результаты обучения	Процедура освоения
ОПК-3 Способен к разработке алгоритмических и программных решений	ОПК-3.1. Знает методы теории алгоритмов, методы системного и	Знает: основы алгоритмизации, принципы разработки	Наблюдение и участие в выполнении упражнений по

<p>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>	<p>прикладного программирования, основные положения и концепции в области математических, информационных и имитационных моделей. ОПК-3.2. Умеет соотносить знания в области программирования, интерпретацию прочитанного, определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем. ОПК-3.3. Имеет практический опыт применения разработки программного обеспечения.</p>	<p>программ</p> <p>Умеет: использовать типы, операции, управляющие структуры и визуальные компоненты, созданные средствами современного языка программирования C#</p> <p>Владеет: навыками создания консольных программ и программ с использованием визуальных средств</p>	<p>системам счисления, самостоятельное и коллективное решение примеров на составление программ. Конспектирование лекций и изучение решенных примеров. Лабораторные и самостоятельные занятия. Ознакомление с разработанными преподавателями кафедры программами для ЭВМ, зарегистрированными в Реестре РФ. Устный опрос</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. ПК-4.2. Умеет реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные</p>	<p>Знает: этапы подготовки программ, подробную структуру программы, простые и структурированные данные, управляющие структуры</p> <p>Уметь: составлять программы средней сложности, воплощать в исполняемые приложения простые базовые алгоритмы</p>	<p>Конспектирование лекций и изучение решенных примеров. Лабораторные и самостоятельные занятия. Демонстрация разработанных преподавателями кафедры программами для ЭВМ, зарегистрированными в Реестре РФ. Устный опрос</p>

	библиотеки, сетевые технологии.	Владеть: навыками компиляции, отладки и тестирования программ		
--	---------------------------------	---	--	--

4. Объем, структура и содержание дисциплины

4.1. Объем дисциплины составляет 3 зачетные единицы, 108 академических часов: 26ч. лекций, 26 ч. лаб., 20 – СРС, 36-контроль.

4.2. Структура дисциплины

Структура и содержание дисциплины «Языки программирования»

№ п/п	Раздел 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)
				Всего	Лек	Пр.	Лаб.	Сам.	Контроль	
1	Модуль 1. Основы языка С#	2								
	Тема 1. Краткий обзор современных средств языка	2	1-2	10	4		4	2		Опрос
	Тема 2. Введение в С# и .Net Framework	2	3	6	2		2	2		Опрос и письменная работа
	Тема 3. Массивы	2	4	6	2		2	1	1	Проверка программ
	Тема 4. Строки и кортежи	2	5	6	2		2	1	1	Проверка программ
	Тема 5. Управляющие структуры	2	6-7	8	4		2	1	1	кол
	Итого по модулю 1			36	14		12	7	3	
	Модуль 2. Развитые средства программирования	2								
	Тема 1. Работа с файлами в С#	2		6	2		2	1	1	

	Тема 2. Современные средства вычислений	2		6	2		2	2		
	Тема 3. Событийно-управляемое и объектно-ориентированное программирование	2		14	4		6	3	1	
	Тема 4. Средства компьютерной графики	2		10	4		4	2		
	Итого по модулю 2	2		36	12		14	8	2	Кол
3	Модуль 3. Подготовка к экзамену	2		36				5	31	
	ВСЕГО	2		108	26		26	20	56	

Очно-заочной и заочной форм обучения для 1 курса по направлению Фиит нет.

4.3. Содержание дисциплины, структурированное по модулям и темам.

4.3.1. Содержание лекционных занятий по дисциплине

Модуль 1. Основы языка C#

Тема 1. Краткий обзор современных средств языка

Действия с другими окнами. Коды окон и их вычисление. Передача кодов нажатых клавиш в окно программы, SendKeys. Запуск стандартных программ. Маршаллинг. Использование Windows Media Player в приложениях Windows Forms. Хронометраж фрагментов программы. Действия с несколькими формами. Работа с веб-страницами. Обращение к посторонним программам (Excel и др.). Организация потоков. Различные типы приложений.

Тема 2. Введение в C# и .Net Framework

Синтаксис: идентификаторы и ключевые слова, литералы, знаки пунктуации и операции. Основы типов: числовые типы и арифметические операции, булевские типы и операции: сравнение и проверка равенств, булевские преобразования, условные операции.

Тема 3. Массивы

Инициализация массива. Неявно типизированные массивы. Определение массива объектов. Свойство Length. Двумерные массивы. Массивы трех и более измерений. Инициализация многомерных массивов. Ступенчатые массивы. Применение свойства Length при обращении со ступенчатыми массивами. Создание массивов. Копирование массивов. Сортировка и поиск. Ковариантность массивов.

Тема 4. Строки и кортежи

Строки. Построение строк. Постоянство строк. Работа со строками. Поле, индекатор и свойство класса String. Операторы класса String. Методы класса String. Сравнение строк в C#.

Кортежи. Символы. Класс Char.

Тема 5. Управляющие структуры

Пустой оператор. Выражения, блоки. Условный оператор if, Оператор выбора switch. Операторы цикла и передачи управления. Цикл с предусловием while. Цикл с постусловием do. Цикл с параметром for. Цикл перебора foreach.

Модуль 2. Развитые средства программирования

Тема 1. Работа с файлами в C#.

Форматированный вывод в C#. Статический метод String.Format и методы экземпляров double.ToString и float.ToString. Цифры после точки. Цифры перед точкой. Разделитель тысяч. Ведущие нули. Выравнивание с помощью пробелов. Пользовательское форматирование отрицательных чисел и нулей.

Пространство имен System.IO и классы для работы с каталогами и дисками. Класс DirectoryInfo для работы с дисками. Работа с каталогами в .NET Framework с применением класса DirectoryInfo. Файлы и классы FileInfo и File. Действия с текстовыми файлами, StreamWriter, FileStream.

Действия с бинарными файлами. Вспомогательный класс File, запись и считывание данных из файла одной строкой. Методы WriteAllText и AppendAllText. Для записи бинарных данных, можно использовать Методы WriteAllBytes, ReadAllText и ReadAllBytes для быстрой записи и быстрого чтения данных.

Тема 2. Современные средства вычислений

Математические и матричные методы. Класс Math и библиотека Matrix. Математические функции. Случайные числа. Методы библиотеки Matrix.

Действия со сверхбольшими числами. Класс BigInteger. Подключение System.Numerics.dll. Способы создания экземпляра объекта BigInteger. Методы класса BigInteger.

Тема 3. Событийно-управляемое и объектно-ориентированное программирование

Создание меню. Создание основного меню. Контекстное меню. Пример создания обработчика события. Пример загрузки изображений.

Визуальные компоненты. Кнопка (Button). Радиокнопка (RadioButton). Окно флажка (CheckBox). Метка (Label). Гиперссылочная метка (LinkLabel). Текстовое окно (TextBox). Расширенное текстовое окно (RichTextBox). Список с помечаемыми окнами (CheckBoxList). Выпадающий список (ComboBox). Список позиций (ListBox). Перемещение по числовому диапазону (NumericUpDown). Текст с предопределённым форматом (MaskedTextBox). Всплывающее окно (ToolTip). Индикатор процесса (ProgressBar). Измерение времени (DateTimePicker, MonthCalendar, компонент Timer).

Деревья и списки. Просмотр списков и деревьев. Действия с ListView. Действия с TreeView. TreeView — иерархическое дерево. Действия с ImageList. Коллекция изображений ImageList. Добавление изображений к списку. Добавление изображений к дереву. Расширенная информация о деревьях.

Тема 4. Средства компьютерной графики

Базовые и развитые средства графики. Методы класса Graphics. Основные понятия интерфейса графических устройств. Независимость приложений от аппаратуры. Контекст отображения. Размещение инструментария. Создание холста -- объекта класса Graphics. Свойства SmoothingMode и Transform. Цвет, перо, кисть, шрифт, точка, прямоугольник. Методы класса Graphics. Рисование графиков. Создание 3-мерных фигур.

Класс GraphicPath. Свойства и методы. Фигурные вырезания и аффинные преобразования. Регионы. Формы произвольной формы (в виде букв, в виде кругов).

4.3.2. Содержание лабораторных занятий по дисциплине

Модуль 1. Основы языка C#

Тема 1. Краткий обзор современных средств языка

Действия с другими окнами. Коды окон и их вычисление. Передача кодов нажатых клавиш в окно программы. Запуск приложений.

Тема 2. Введение в C# и .Net Framework

Синтаксис: идентификаторы и ключевые слова, литералы, знаки пунктуации и операции. Основы типов и операции.

Тема 3. Массивы. Инициализация массива. Двумерные массивы. Ступенчатые массивы. Сортировка и поиск.

Тема 4. Строки и кортежи. Строки: построение и свойство постоянства строк; свойства, операторы и методы класса String.

Тема 5. Управляющие структуры. Операторы: пустой, выражения, условный, перехода, выбора, цикл с предусловием while, цикл с постусловием do, цикл с параметром for, цикл перебора foreach.

Модуль 2. Развитые средства программирования

Тема 1. Работа с файлами в C#. Работа с файлами в C#: форматированный вывод в C#, пространство имен System.IO и классы для работы с каталогами и дисками, действия с текстовыми файлами, действия с бинарными файлами.

Тема 2. Современные средства вычислений. Класс Math и библиотека Matrix. Математические функции. Класс BigInteger.

Тема 3. Событийно-управляемое и объектно-ориентированное программирование. Создание меню. Создание основного меню. Контекстное меню. Пример создания обработчика события. Пример загрузки изображений. Визуальные компоненты.

Тема 4. Средства компьютерной графики.

Средства компьютерной графики: свойства и методы классов Graphics и GraphicPath.

5. Образовательные технологии

5.1. Процесс изложения учебного материала сопровождается систематическими (на каждом занятии) компьютерными презентациями лекционного материала, кода и результатов программ, а также демонстрацией решения задач с использованием мультимедийного оборудования.

5.2. Студентам регулярно сообщаются те из формулировок задач, предложенных на собеседовании в ведущие IT-компании (Microsoft, Apple, Google, Uber, Amazon.com и др.), которые соответствуют текущим темам.

5.3. Предусмотрено регулярное общение и консультации с представителями российских и зарубежных компаний (из числа выпускников кафедры) по электронной почте и по скайпу.

5.4. При подготовке к занятиям преподаватели кафедры учитывают материалы лекций в некоторых ведущих вузах страны (например, на факультете информатики МГТУ им. Баумана).

6. Учебно-методическое обеспечение самостоятельной работы студентов

Для обеспечения самостоятельной работы используется разработанный на кафедре пакет заданий и методических указаний, изданы три учебных пособия с заданиями и программами.

6.1 Примерное распределение времени самостоятельной работы студентов

Вид самостоятельной работы	Примерная трудоёмкость, а.ч.
работа с лекционным материалом	6
самостоятельное изучение разделов дисциплины	4
подготовка к лабораторным работам	6
Творческая проблемно-ориентированная СРС	
исследовательская работа, участие в конференциях и интернет-олимпиадах	4
Итого СРС:	20

6.2. Виды и порядок выполнения самостоятельной работы

1. Изучение презентационных материалов и рекомендованной литературы.
2. Выполнение индивидуального пакета заданий на составление программ.
3. Решение задач и упражнений, изложенных на лекции.
4. Подготовка к отчету по индивидуальному пакету заданий
5. Подготовка к экзамену

6.3. Порядок контроля: 1. Блиц-опрос на лабораторных занятиях, 2. Проверка выполнения пакета заданий и прием отчета по ним, 3. Текущий контроль за выполнением задач, сформулированных на лекции, 4. Коллоквиумы, 5. Экзамен.

Раздел (модуль. тема)	Вид самостоятельной работы, практическое содержание	Контрольные сроки (в нед.) и вид контроля	Уч.-мет.обеспечение (указаны источники из списка основной

			литературы)
1.1	Введение в C#	1 (Устной опрос)	[2], 33-40
1.2	Числовые и логические типы	2(устный опрос)	[2], с. 54-63 [3], с. 284-285
1.3	Действия с массивами	3 (письменный опрос) Коллоквиум	[2], 65-68 [3], с. 209-220
1.4	Строки. Свойства и методы	4-5 (проверка программ по домашним заданиям)	[2], 63- 65 [3], с.285-288
1.5	Операторы	6-7 (проверка выполнения компьютерных программ)	[2], с. 65-68
2.1	Работа с файлами	8 (проверка пакета заданий)	[3], 61-85, 291-293
2.2	Современные средства вычислений	9 (проверка примеров с длинной арифметикой)	[3], 294-396
2.3	Визуальные компоненты.	10-11 (отчет по графическим компонентам)	[3], 249-277
2.4	Средства компьютерной графики	12-13 (отчет по графическим проектам)	[3], 86-120, 277-283

Текущий контроль:

1. Проверка программ на языке C# по заданиям индивидуального пакета;
2. Проверка изучения лекционного материала;
3. Промежуточная аттестация в форме отчета по пакетам заданий.

Текущий контроль включает систематический блиц-опрос и проверку выполнения компьютерных программ.

Промежуточный контроль проводится в виде отчета по пакетам заданий, предварительная проверка решений практикуется по файлам, отправленным по электронной почте.

Итоговый контроль проводится в виде письменной экзаменационной работы с обязательным устным собеседованием по результатам предварительной проверки письменной работы.

Критерии выставления оценок «отлично», «хорошо», «удовлетворительно» определяются не только степенью владения материалом и практическими навыками составления программ, но также (с учетом низкого уровня школьной подготовки в регионе) и динамикой знаний и навыков студента за оцениваемый период.

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 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

(Указываются темы эссе, рефератов, курсовых работ и др. Приводятся примерные тестовые задания, контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины.)

7.1.1. Темы курсовых работ и рефератов:

Консольные приложения Delphi и C#. Сравнительная характеристика.

Компьютерный вывод рекуррентных формул (на примере задачи разбиения).

Длинная целочисленная арифметика.

Числа Фибоначчи. Вычислительные аспекты.

Настольные графические приложения.

Проекты с несколькими формами.

Компьютерная графика в C#-проектах.

OpenGL в проектах C#.

Проекты с применением веб-камер.

Распознавание цифр.

Визуальные компоненты Visual C#.

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 (одно лабораторное занятие)

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. Примерный перечень вопросов текущего контроля (по материалам одной лекции)

1. Перечислите компоненты для организации диалога.

2. Как выбрать файл, используя `openFileDialog`? Как выбрать каталог, используя `folderBrowserDirectory`?

3. Смысл свойства InitialDirectory. Как записать в это свойство значение текущего каталога.
4. Объясните действия условного оператора вида
if (Объект.ShowDialog() == DialogResult.OK) {оператор}
5. Объясните 5-6 основных методов класса Directory.
6. Создание и удаление подкаталога.
7. Как вывести все подкаталоги и все файлы заданного каталога?
8. Вывести информацию о заданном каталоге, используя DirectoryInfo.
9. Преимущество списков перед массивами.
10. Создание списка типа List.
11. Объясните 5-6 основных методов списка List.
12. Список с разнотипными объектами. Класс ArrayList.
13. Создание и пример использования строки состояния.
14. Создание и пример использования ListBox.
15. Создание и пример использования ComboBox.
16. Создание и пример использования CheckedList.

7.1.5. Примерные вопросы к промежуточному контролю (коллоквиуму)

Напишите ряд Тейлора для функции $y = \sin x$. Выведите сумму первых n слагаемых ряда. Значения x и n задайте по своему усмотрению.

Напишите ряд Тейлора для функции $y = \cos x$. Найдите сумму ряда с точностью до ϵ . Значение ϵ задайте по своему усмотрению.

Введите целое число в `textBox1`, при нажатии на кнопку выведите все его простые делители в `listBox1`.

Создайте две компоненты `Listbox`, соединенные (на форме) разнонаправленными стрелками. При нажатии на соответствующую стрелку строки из одной компоненты должны переместиться в другую.

Задан текстовый файл `1.txt`, каждая строка которого содержит фамилию и имя студентов (разделитель – пробел). Создайте `ComboBox` с 3 фамилиями. При выборе любой из них программа выводит в строку статуса соответствующее имя.

Задайте элементы двух квадратных матриц $n \times n$ случайным образом. Найдите произведение матриц. Используйте `DataGridView` для рисования.

Элементы заданного квадратного массива упорядочить по возрастанию. Выяснить, имеется ли среди них значение n , если ДА, то в какой позиции. n задайте по своему усмотрению.

В заданном одномерном массиве с целочисленными элементами найдите наименьший и наибольший элементы, наименьший элемент замените их НОД, а наибольший – НОК.

Запуск мультимедийных приложений. Элемент управления `WindowsMediaPlayer`. Использование `DataGridView`, его свойства.

Составьте программу замедленного рисования графика функции $y=\sin x$. При нажатии на кнопку рисование начинается. При следующем нажатии прекращается. При очередном нажатии продолжается рисование. При четвертом нажатии снова прекращается и т.д.

7.1.6. ВОПРОСЫ К ЭКЗАМЕНУ

1. Вызов внешних функций и взаимодействие с внешними приложениями. Какое пространство имен подключить? Как использовать `DllImport`? Что такое маршаллинг?

A1. Удалить из папки `C:\SomeDir` все файлы с расширением `bmp`.

2. Запуск внешних приложений. Поиск и активизация окна. Вычисление дескриптора окна. Передача кодов нажатых клавиш.

A2. Текущий каталог `A` программы вложен в каталог `B`, содержащий, в частности, текстовый файл `1.txt` с фамилиями студентов. При нажатии на кнопку программа должна вывести все фамилии в компоненту `ListBox`.

3. Массивы. Объявление и инициализация. Оператор цикла для доступа к элементам массива.

A3. Программа должна удалить из выбранной папки все файлы блокнота ("`.txt`"). Если имя папки завершается знаком "`!`", вывести предупреждение о последствиях.

4. Многомерные массивы. Основные методы класса `System.Array`.

A4. В текущем каталоге размещен текстовый файл `1.txt` с русским текстом. Введите слово в компоненту `textBox1`. Все строки файла, содержащие данное слово, отобразите в `listBox1`. Выделите произвольную строку в `listBox1`. При нажатии на кнопку три строки файла до выделенной строки и три после нее должны быть выведены в `listBox2`.

5. Строки. Примеры объявления и создания. Основные методы класса `String`.

A5. В текущем каталоге размещен текстовый файл `1.txt`, содержащий целое `n`. Найдите сумму его цифр и поместите в файл `2.txt`.

6. Операции со строками. Склеивание с разделителями: `String.Concat`, `String.Join`. Сравнение: `String.Compare`, `String.Equals`. Поиск: `s.IndexOf`, `s.LastIndexOf`, `s.StartsWith`, `s.EndsWith`. Разрезание строки: `s.Split`. Обрезка: `s.Trim`, `s.TrimStart`, `s.TrimEnd`, `s.Substring`. Вставка подстроки с заданного места: `s.Insert`. Удаление подстроки: `s.Remove`. Замена всех подстрок: `s.Replace`. Преобразование к одному рег.: `s.ToUpper()`, `s.ToLower()`.

A6. Напишите ряд Тейлора для функции $y=\sin x$. Выведите сумму первых `n` слагаемых ряда. Значения `x` и `n` задайте по своему усмотрению.

7. Диалоговые окна, компоненты `openFileDialog1`, `saveFileDialog1`, `folderBrowserDialog1`, `fontDialog1`, `colorDialog1` и действия с ними. Метод `ShowDialog`.

A7. Напишите ряд Тейлора для функции $y=\cos x$. Найдите сумму ряда с точностью до `epsilon`. Значение `epsilon` задайте по своему усмотрению.

8. Методы и свойства класса `Directory`. Начальный и текущий каталоги. Создание каталога, проверка существования и уничтожение. Родительский каталог. Коллекция файлов и каталогов из выбранного каталога.

A8. Введите целое число в `textBox1`, при нажатии на кнопку выведите все его простые делители в `listBox1`.

9. Однотипные списки. Методы: добавление элемента (массива, коллекции), сортировка, поиск, вставка, удаление.

A9. Создайте две компоненты `ListBox`, соединенные (на форме) разнонаправленными стрелками. При нажатии на соответствующую стрелку строки из одной компоненты должны переместиться в другую.

10. Класс `ArrayList`. Форматированный вывод.

A10. Используя `TreeView`, нарисуйте двоичное дерево для алгоритма Хаффмана, если входное сообщение имеет вид: AAABCCCCDD. В `ListBox` укажите код для каждой буквы.

11. Основные типы языка C#, стандартные операции.

A11. Создайте меню с двумя пунктами: Выход и Каталоги. При нажатии на Выход программа завершается. При нажатии на Каталоги выводит в файл список подкаталогов текущего каталога. Программа содержит также строку статуса из двух панелей, в первую выводятся координаты курсора мыши, во вторую – дата.

12. Основные управляющие структуры языка (операторы).

A12. Задан текстовый файл `1.txt`, каждая строка которого содержит фамилию и имя студентов (разделитель – пробел). Создайте `ComboBox` с 3 фамилиями. При выборе любой из них программа выводит в строку статуса соответствующее имя.

13. Создание основного и контекстного меню. Хронометраж выполнения фрагментов программы.

A13. Используйте компоненты `openFileDialog1`, `colorDialog1`, `fontDialog1`, чтобы вывести имя выбранного интерактивно файла заданным цветом и заданным шрифтом в том месте формы, где производится щелчок кнопкой мыши.

14. Регионы. Формы непрямоугольных очертаний.

A14. Задайте элементы двух квадратных матриц $n \times n$ случайным образом. Найдите произведение матриц. Используйте `DataGridView` для рисования.

15. Использование компонент `MenuStrip`, `StatusStrip`, `ComboBox`, `ListBox`, `CheckedListBox`, создание соответствующих элементов управления.

A15. Элементы заданного квадратного массива упорядочить по возрастанию. Выяснить, имеется ли среди них значение n , если ДА, то в какой позиции. n задайте по своему усмотрению.

16. Компонента `ListView`. Свойство `View` и его значения. Действия с компонентой `ListView`, выделенные элементы. Применение `ImageList`.

A16. В заданном одномерном массиве с целочисленными элементами найдите наименьший и наибольший элементы, наименьший элемент замените их НОД, а наибольший – НОК.

17. Компонента `TreeView`. Тип `TreeNode`. Свойство `Nodes`. Выделенные узлы. Добавление узлов. Применение `ImageList`.

A17. На форме размеров 600×600 должны быть созданы 10 рядов квадратных кнопок размерами 30×30 по 10 кнопок в ряду.

18. Запуск мультимедийных приложений. Элемент управления `WindowsMediaPlayer`. Использование `DataGridView`, его свойства.

A18. Проведите линии, образующие сетку, напоминающую фрагмент клеточной тетрадки. В одну из клеток поместите закрашенный кружок. При щелчке по клетке кружок должен переместиться в нее.

19. Основные понятия интерфейса графических устройств. Независимость приложений от аппаратуры. Контекст отображения. Размещение инструментария. Класс Graphics. Создание холста -- объекта класса Graphics. Цвет, перо, кисть, шрифт, точка, прямоугольник.

A19. Составьте программу медленного рисования графика функции $y = \sin x$. При нажатии на кнопку рисование начинается. При следующем нажатии прекращается. При очередном нажатии продолжается рисование. При четвертом нажатии снова прекращается и т.д.

20. Класс GraphicsPath. Аффинные преобразования.

A20. Составьте программу, где вокруг закрашенного круга радиуса 20 с разными скоростями и по разным эллипсоидальным орбитам вращаются три других закрашенных круга с радиусом 10. При нажатии на кнопку вращение прекращается, при повторном нажатии продолжается.

21. Основные пункты схемы рисования графиков функции одной и двух переменных.

A21. Составьте программу для вращения звезды с n лучами (n выбирается из numericUpDown). При нажатии на кнопку вращение завершается.

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

Общий результат складывается из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- посещение занятий – 10 баллов,
- выполнение текущих лабораторных заданий – 40 баллов,
- выполнение домашних (аудиторных) контрольных работ - 50 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 50 баллов,
- письменная контрольная работа - 50 баллов.

8. Учебно-методическое обеспечение дисциплины

а) <http://cathedra.dgu.ru/EducationalProcess.aspx?Value=18&id=6>

б) Основная:

1. Биллиг В.А. Основы программирования на C# [Электронный ресурс]/ Биллиг В.А.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 574 с.— Режим доступа: <http://www.iprbookshop.ru/73695.html>.— ЭБС «IPRbooks»

2. Эндрю Троелсен. Язык программирования C# 5.0 и платформа .NET 4.5 (6-е издание). Издательство: Вильямс, 2013.

3. Джозеф Албахари, Бен Албахари. C# 5.0. Справочник. Полное описание языка (5-е издание). Издательство: Вильямс, 2013. - 1054 стр.

в) Дополнительная:

1. Кариев Ч.А. Разработка Windows-приложений на основе Visual C# [Электронный ресурс]: учебное пособие/ Кариев Ч.А.— Электрон. текстовые данные.— Москва,

Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017.— 768 с.— Режим доступа: <http://www.iprbookshop.ru/72340.html>.— ЭБС «IPRbooks»

2. Снетков В.М. Практикум прикладного программирования на С# в среде VS.NET 2008 [Электронный ресурс]/ Снетков В.М.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 1691 с.— Режим доступа: <http://www.iprbookshop.ru/62823.html>.— ЭБС «IPRbooks»

3. Павловская Т.А. Программирование на языке высокого уровня С# [Электронный ресурс]/ Павловская Т.А.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 245 с.— Режим доступа: <http://www.iprbookshop.ru/73713.html>.— ЭБС «IPRbooks»

4. Лабор В.В. Си Шарп: Создание приложений для Windows. – Мн.: Харвест, 2011. – 384 с.

9. Перечень рекомендуемых ресурсов информационно-телекоммуникационной сети «Интернет»

Википедия // URL: https://ru.wikipedia.org/wiki/C_Sharp (дата просмотра: 10.11. 2019).

Уроки программирования с нуля. Си-шарп для чайников // URL: <http://mycsharp.ru> (дата просмотра: 10.11.2019).

С# для начинающих // URL: <http://www.cyberforum.ru/csharp-beginners/> (дата просмотра: 11.11.2019).

10. Методические указания для обучающихся по освоению дисциплины

1) В профессиональной деятельности программиста выполнение проекта, как правило, представляет сплав алгоритмических и программистских решений. Поэтому языковые возможности С# следует систематически применять к задачам по дискретной математике – дисциплине, изучение которой происходит синхронно с преподаванием данной дисциплины.

2) Помимо выполнения лабораторных заданий рекомендуется решить все упражнения (или большую часть) упражнений, предложенных к каждой лекции.

3) Упражнения по первому модулю рекомендуется решить как на языке С#, так и на языке Delphi.

4) Выбор структур для представления исходных данных особенно важен в тех случаях, когда в задании имеются требования к оценке сложности алгоритма.

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

Операционная система Windows 7, 8.1 и 10, среда программирования MS Visual Studio.

Кроме того, на отдельных занятиях привлекается специализированное программное обеспечение (например, на занятиях со сверхбольшими числами -система компьютерной математики Mathematica, на занятиях с окнами приложений – утилиты вычисления дескрипторов).

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

На каждой лекции используется стационарное мультимедийное презентационное оборудование (ауд. 3-72). Часть лекций предоставляется студенту в электронном формате.

Лабораторные занятия проводятся в компьютерных классах с современным аппаратным и программным обеспечением. При выполнении лабораторных заданий студенту предоставляется право выбора одного из двух языков программирования из поддерживаемых MS Visual Studio. На сайте кафедры размещаются учебные пособия и презентации к лекции.