

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Белгородский Валерий Савельевич
Должность: Ректор
Дата подписания: 11.01.2024 12:36:51
Уникальный программный ключ:
8df276ee93e17c18e7bee9e7cad2d0ed9ab82473

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Российский государственный университет им. А.Н. Косыгина
(Технологии. Дизайн. Искусство)»

Институт Информационных технологий и цифровой трансформации
Кафедра Прикладной математики и программирования

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Алгоритмы и структуры данных

Уровень образования	бакалавриат
Направление подготовки	01.03.02 Прикладная математика и информатика
Направленность (профиль)	Системное программирование и компьютерные технологии
Срок освоения образовательной программы по очной форме обучения	4 года
Форма обучения	очная

Рабочая программа учебной дисциплины «Алгоритмы и структуры данных» основной профессиональной образовательной программы высшего образования, рассмотрена и одобрена на заседании кафедры, протокол № 6 от 14.02.2023 г.

Разработчики рабочей программы учебной дисциплины:

1. Преподаватель А.Т. Костоев
 2. Доцент А.М. Романенков
- Заведующий кафедрой: О.П. Новиков

1. ОБЩИЕ СВЕДЕНИЯ

Учебная дисциплина «Алгоритмы и структуры данных» изучается в третьем семестре. Курсовая работа не предусмотрена.

1.1. Форма промежуточной аттестации:

Экзамен.

1.2. Место учебной дисциплины в структуре ОПОП

Учебная дисциплина «Алгоритмы и структуры данных» относится к обязательной части.

Основой для освоения дисциплины являются результаты обучения по предшествующей дисциплине:

- Стандартные методы программирования на языках высокого уровня;
- Языки и методы программирования.

Результаты обучения по учебной дисциплине, используются при изучении следующих дисциплин:

- Алгоритмы анализа данных;
- Проектирование баз данных и информационных систем;
- Информационная безопасность;
- Операционные системы и системное программирование.

Результаты освоения учебной дисциплины в дальнейшем будут использованы при прохождении производственной практики и (или) выполнении выпускной квалификационной работы.

2. ЦЕЛИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Целями изучения дисциплины «Алгоритмы и структуры данных» являются:

- углубленное понимание различных алгоритмов и структур данных, их свойств и принципов работы на языке C#;
- развитие навыков разработки эффективных алгоритмов на языке C#;
- развитие практических навыков решения задач с использованием алгоритмов и структур данных на языке C#;
- развитие практических навыков решения задач с использованием алгоритмов и структур данных на языке C#;
- развитие навыков анализа сложности алгоритмов на языке C#;
- изучение алгоритмов и структур данных на языке C#;
- формирование у обучающихся компетенций, установленных образовательной программой в соответствии с ФГОС ВО по данной дисциплине.

Результатом обучения по учебной дисциплине является овладение обучающимися знаниями, умениями, навыками и опытом деятельности, характеризующими процесс формирования компетенций и обеспечивающими достижение планируемых результатов освоения учебной дисциплины.

2.1. Формируемые компетенции, индикаторы достижения компетенций, соотнесённые с планируемыми результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
ОПК-4 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ИД-ОПК-4.3 Выбор и применение основных структур данных для построения информационных систем	<ul style="list-style-type: none"> – Формулирует алгоритм решения задачи в виде последовательности действий, выражаемой на языке программирования С#; – Понимание основных алгоритмов и структур данных; – Умение анализировать и оценивать алгоритмы; – Разработка и реализация алгоритмов на языке С#;
ПК-2 Способен разрабатывать и адаптировать прикладное программное обеспечение	ИД-ПК-2.3 Применение и реализация математически сложных алгоритмов в современных программных комплексах	<ul style="list-style-type: none"> – Применение алгоритмов и структур данных для решения задач; – Анализирует и оптимизирует программный код;
ПК-3 Способен участвовать в создании, модификации и сопровождении информационных систем	ИД-ПК-3.3 Использование основных алгоритмических и программных решений, связанных с объектами и системами информационных технологий	<ul style="list-style-type: none"> – Понимание принципов программирования; – Реализует алгоритмы решения выбранной задачи на языке программирования С#;
ПК-5 Способен осуществлять анализ больших данных с использованием существующих методов, технологий и инструментов	ИД-ПК-5.1 Использование математических и аналитических методов исследования данных	<ul style="list-style-type: none"> – Оценивает сложность работы алгоритма; – Анализирует основные способы отладки и профилировки программ, разработанных на платформе Net; – Проводит профилирование и отладку программного кода; – Тестирует производительность и безопасность программ, разработанных на платформе Net; – Анализирует основные языковые конструкции С#, средства разработки (компиляторы, IDE, отладчики и т.д.); – Разрабатывает приложения для выбранной задачи на языке программирования С#; – Проектирует программы с использованием концепций объектно-ориентированного программирования.

3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Общая трудоёмкость учебной дисциплины по учебному плану составляет:

по очной форме обучения –	4	з.е.	144	час.
---------------------------	---	------	-----	------

3.1. Структура учебной дисциплины для обучающихся по видам занятий

Структура и объем дисциплины									
Объем дисциплины по семестрам	форма промежуточной аттестации	всего, час	Контактная аудиторная работа, час				Самостоятельная работа обучающегося, час		
			лекции, час	практические занятия, час	лабораторные занятия, час	практическая подготовка, час	курсовая работа	самостоятельная работа обучающегося, час	промежуточная аттестация, час
3 семестр	экзамен	144	34		34			40	36
Всего:		144	34		34			40	36

3.2. Структура учебной дисциплины для обучающихся по разделам и темам дисциплины:

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы, час	Практическая подготовка, час		
ОПК-4 ИД-ОПК-4.3; ПК-2 ИД-ПК-2.3; ПК-3 ИД-ПК-3.3; ПК-5 ИД-ПК-5.1	Раздел I						
	Тема 1.1 Общие понятия	2				2	Формы текущего контроля: лабораторные работы
	Тема 1.2 Приложение	2				2	
	Тема 1.3 Пространство имён. Сборка. Класс	2				2	
	Тема 1.4 Синтаксис	2				2	
	Тема 1.5 Управление выполнением	2				2	
	Тема 1.6 Классы и объекты	2				4	
	Тема 1.7 Перегруженные операции	2				4	
	Лабораторная работа №1 Среда разработки Visual Studio			2		3	
	Лабораторная работа №2 Линейные алгоритмы			2		3	
	Лабораторная работа №3 Разветвляющиеся алгоритмы			2		3	
	Лабораторная работа №4 Циклические алгоритмы			2		3	
	Лабораторная работа №5 Классы и объекты			2		3	
	Лабораторная работа №6			4		3	

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы, час	Практическая подготовка, час		
	Строки						
	Раздел II						
ОПК-4 ИД-ОПК-4.3; ПК-2 ИД-ПК-2.3; ПК-3 ИД-ПК-3.3; ПК-5 ИД-ПК-5.1	Тема 2.1 Наследование и полиморфизм	2				3	Формы текущего контроля: лабораторные работы
	Тема 2.2 Параметризованные классы	2				3	
	Тема 2.3 Методы, интерфейсы и делегаты	2				3	
	Тема 2.4 Ввод и вывод	2				3	
	Тема 2.5 Language Integrated Query - LINQ	4				3	
	Тема 2.6 Атрибуты, сборки, рефлексия	4				3	
	Тема 2.7 Форма	4				3	
	Лабораторная работа №7 Одномерные массивы			2		3	
	Лабораторная работа №8 Многомерные массивы			2		3	
	Лабораторная работа №9 Графики функций			4		3	
	Лабораторная работа №10 Методы			4		3	
	Лабораторная работа №11 Рекурсия			4		3	

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы, час	Практическая подготовка, час		
	Лабораторная работа №12 Сортировка и поиск			4		4	
	Экзамен	х	х	х	х	36	Экзамен – устный опрос по темам дисциплины
	ИТОГО за третий семестр	34		34		3	
	ИТОГО за весь период	53		34		40	

3.3. Краткое содержание учебной дисциплины

№ пп	Наименование раздела и темы дисциплины	Содержание раздела (темы)
Первый раздел		
Тема 1.1	Общие понятия	Принципы ООП; обзор .NET. Обзор и версии языка C#.
Тема 1.2	Приложение	Средства разработки, приложения .NET. Точка входа в приложение; консольное приложение. Приложение Windows Form. Приложение wfp.
Тема 1.3	Пространство имён. Сборка. Класс	Класс и структура. Система типов. Категории типов. Object и string: предопределённые ссылочные типы. Приведение типов. Пространство имен. Параметризованные типы.
Тема 1.4	Синтаксис	Литералы: представление значений. Переменные элементарных типов: объявление и инициализация. Константы. Операции и выражения. Приоритет операций. Особенности выполнения арифметических операций. Особенности арифметики с плавающей точкой: checked и unchecked. Контроль за переполнением. Константное выражение. Перечисления. Объявление переменных. Область видимости и время жизни.
Тема 1.5	Управление выполнением	Управляющие операторы. Методы. Перегрузка методов. Обработка исключений. Способы передачи параметров при вызове метода. Ссылка и ссылка на ссылку как параметры. Сравнение значений ссылок. Свойства: открытые, закрытые, автоматически реализуемые.
Тема 1.6	Классы и объекты	Конструктор. Операция new. Создание объекта. Конструктор умолчания. This в конструкторе, деструкторе, методе, свойстве. This в заголовке конструктора. Средства инициализации (явные и упрощённые, именованные параметры инициализации). Уничтожение объектов в управляемой памяти. Деструктор. Метод finalize. Класс gc. Массивы: объявление и инициализация. Типы массивов. Встроенный сервис по обслуживанию простых массивов. Массивы как параметры. Спецификатор params. Value type, reference type. Неявно типизированные локальные переменные, статические поля и конструкторы. Анонимные классы.
Тема 1.7	Перегруженные операции	Перегрузка операций. Операторная функция. Унарные операторные функции. Бинарные операторные функции. Operator true и operator false. Определение операций. Конъюнкция и дизъюнкция. Операции и &&. Свойства и индексы. Explicit и implicit: преобразования явные и неявные.
Второй раздел		
Тема 2.1	Наследование и полиморфизм	Явное обращение к конструктору базового класса. Переопределение членов базового класса. Наследование и new-модификатор. Полное квалифицированное имя. Sealed-спецификатор: прекращение наследования. Абстрактные функции и абстрактные классы. Ссылка на объект базового класса. Операции is и as. Приведение к типу object: boxing и unboxing. Виртуальные функции. Принцип полиморфизма. Интерфейсы: объявление интерфейса. Реализация сортировки в массиве. Интерфейс comparable. Интерфейс comparer. Интерфейс

		idisposable: освобождение ресурсов. Делегаты и события. Перегрузка методов базовых классов (функция <code>ToString()</code>).
Тема 2.2	Параметризованные классы	Коллекции. Перечислитель. Шаблоны. Шаблоны классов и функций. Ограничения параметра типа. Пример использования шаблонов: сортировка. Шаблоны классов и функций. Ограничения параметра типа. Nullable типы. Обзор nullable типов. Nullable types: члены класса. Явное и неявное преобразование. Операции в nullable types. Операция <code>??</code> . Тип <code>bool?</code> , статическая и динамическая типизация.
Тема 2.3	Методы, интерфейсы и делегаты	Анонимные методы, делегаты, делегаты <code>Action<></code> , <code>Function<></code> , <code>Comparison<></code> , <code>Predicate<></code> . Методы расширения. Лямбда-выражения.
Тема 2.4	Ввод и вывод	Потоки ввода-вывода. Общая характеристика классов потоков. Потоки: байтовые, символьные, двоичные. Предопределенные потоки ввода-вывода. Консольный ввод-вывод. Функции-члены класса <code>console</code> . Консольный вывод; форматирование; нестандартное (<code>custom</code>) форматирование значений; консольный ввод; преобразование значений; файловый ввод-вывод.
Тема 2.5	Language Integrated Query - LINQ	Расширения для LINQ: лямбда-выражения, переменная <code>var</code> , инициализаторы объектов и коллекций, анонимные типы, методы расширения, частичные (<code>partial</code>) методы, выражения запросов. Интерфейс <code>IEnumerable<T></code> . Последовательности, стандартные запросы. Операции <code>OfType</code> и <code>Cast</code> . Отложенные запросы. <code>Objects</code> , <code>LINQ to Objects</code> . <code>XML</code> , <code>LINQ to XML</code> . <code>SQL</code> , <code>LINQ to SQL</code> .
Тема 2.6	Атрибуты, сборки, рефлексия	Пространство имен <code>System.Reflection</code> . Класс <code>System.Type</code> . Реализация отражения: <code>Type</code> , <code>InvokeMember</code> , <code>BindingFlags</code> . Атрибуты. Сборка: класс <code>Assembly</code> . Версия сборки. Файл конфигурации приложения. Общедоступная сборка. Сборки из <code>GAC</code> . Динамические сборки. Создание, сохранение, загрузка и выполнение сборки.
Тема 2.7	Форма	Управление и события жизненного цикла. Контейнер как элемент управления. Элементы управления и компоненты. Свойства элементов управления: <code>Anchor</code> и <code>Dock</code> . <code>Extender providers</code> . Провайдеры дополнительных свойств. <code>Validating</code> и <code>Validated</code> элементов управления. Управление посредством сообщений. Стандартный делегат. Делегат <code>EventHandler</code> . Класс <code>Application</code> . События класса <code>Application</code> . <code>Windows message</code> . Примеры перехвата сообщений. Метод <code>WndProc</code> . Пример переопределения <code>WndProc</code> . Контекст приложения. <code>GDI+</code> . <code>Graphics</code> . Битовая карта как поверхность для рисования. <code>GraphicsPath</code> . <code>Region</code> . Применение классов <code>GraphicsPath</code> и <code>Region</code> . Собственные элементы управления.

3.4. Организация самостоятельной работы обучающихся

Самостоятельная работа студента – обязательная часть образовательного процесса, направленная на развитие готовности к профессиональному и личностному самообразованию, на проектирование дальнейшего образовательного маршрута и профессиональной карьеры.

Самостоятельная работа обучающихся по дисциплине организована как совокупность аудиторных и внеаудиторных занятий и работ, обеспечивающих успешное освоение дисциплины.

Аудиторная самостоятельная работа обучающихся по дисциплине выполняется на учебных занятиях под руководством преподавателя и по его заданию. Аудиторная самостоятельная работа обучающихся входит в общий объем времени, отведенного учебным планом на аудиторную работу, и регламентируется расписанием учебных занятий.

Внеаудиторная самостоятельная работа обучающихся – планируемая учебная, научно-исследовательская, практическая работа обучающихся, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия, расписанием учебных занятий не регламентируется.

Внеаудиторная самостоятельная работа обучающихся включает в себя:

- изучение теоретического и практического материала по рекомендованным источникам;
- выполнение лабораторных работ;
- подготовка к промежуточной аттестации в течение семестра.

3.5. Применение электронного обучения, дистанционных образовательных технологий

При реализации программы учебной дисциплины электронное обучение и дистанционные образовательные технологии не применяются.

4. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, КРИТЕРИИ ОЦЕНКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ, СИСТЕМА И ШКАЛА ОЦЕНИВАНИЯ

4.1. Соотнесение планируемых результатов обучения с уровнями сформированности компетенции.

Уровни сформированности компетенций	Итоговое количество баллов в 100-балльной системе по результатам текущей и промежуточной аттестации	Оценка в пятибалльной системе по результатам текущей и промежуточной аттестации	Показатели уровня сформированности	
			Общепрофессиональной компетенции	профессиональной компетенции
			ОПК-4 ИД-ОПК-4.3; ПК-2 ИД-ПК-2.3; ПК-3 ИД-ПК-3.3; ПК-5 ИД-ПК-5.1	ОПК-4 ИД-ОПК-4.3; ПК-2 ИД-ПК-2.3; ПК-3 ИД-ПК-3.3; ПК-5 ИД-ПК-5.1
высокий		отлично	<p>Обучающийся:</p> <ul style="list-style-type: none"> – исчерпывающе и логически стройно излагает учебный материал, умеет связывать теорию с практикой, справляется с решением задач профессиональной направленности высокого уровня сложности, правильно обосновывает принятые решения; – на высоком уровне реализует алгоритмы решения выбранной задачи на языке программирования С#; – демонстрирует высокий уровень разработки приложений для выбранной задачи на языке программирования С#; – на высоком уровне осуществляет профилирование и отладку программного кода; – свободно ориентируется в учебной и профессиональной литературе; – дает развернутые, исчерпывающие, профессионально грамотные ответы на вопросы, в том числе, дополнительные. 	
повышенный		хорошо	<p>Обучающийся:</p> <ul style="list-style-type: none"> – достаточно подробно, грамотно и по существу излагает изученный материал, приводит и раскрывает в тезисной форме основные понятия; – достаточно хорошо реализует алгоритмы решения выбранной задачи на языке программирования С#; – демонстрирует достаточно хороший уровень разработки приложений для выбранной задачи на языке программирования С#; 	

			<ul style="list-style-type: none"> – на достаточно хорошем уровне осуществляет профилирование и отладку программного кода; – достаточно хорошо ориентируется в учебной и профессиональной литературе; – дает ответы на поставленные вопросы, отражающие знания теоретического материала, при этом, не допуская существенных неточностей.
базовый		удовлетворительно	<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует теоретические знания основного учебного материала дисциплины в объеме, необходимом для дальнейшего освоения образовательной программы; – с некоторыми затруднениями реализует алгоритмы решения выбранной задачи на языке программирования С#; – демонстрирует базовый уровень разработки приложений для выбранной задачи на языке программирования С#; – с затруднениями осуществляет профилирование и отладку программного кода; – дает ответы, отражающие знания на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профилю обучения.
низкий		неудовлетворительно	<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует фрагментарные знания теоретического и практического материала, допускает грубые ошибки при его изложении на занятиях и в ходе промежуточной аттестации; – испытывает серьезные затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, не владеет необходимыми для этого навыками и приемами; – реализует алгоритмы решения выбранной задачи на языке программирования С# только при участии преподавателя; – демонстрирует низкий уровень разработки приложений для выбранной задачи на языке программирования С#; – не способен осуществлять профилирование и отладку программного кода; – дает ответы, отражающие отсутствие знаний на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы.

5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ, ВКЛЮЧАЯ САМОСТОЯТЕЛЬНУЮ РАБОТУ ОБУЧАЮЩИХСЯ

При проведении контроля самостоятельной работы обучающихся, текущего контроля и промежуточной аттестации по учебной дисциплине «Алгоритмы и структуры данных» проверяется уровень сформированности у обучающихся компетенций и запланированных результатов обучения по дисциплине, указанных в разделе 2 настоящей программы.

5.1. Формы текущего контроля успеваемости, примеры типовых заданий:

№ пп	Формы текущего контроля	Примеры типовых заданий
1	Лабораторные работы	<ol style="list-style-type: none"> 1. Напишите две функции для поиска всех корней кубического уравнения двумя разными способами: с помощью формулы Кордано и без нее. Результат возвращайте через заголовок функции. Сравните полученные результаты. 2. Напишите функции, вычисляющие значения чисел e, π, $\ln 2$, $\sqrt{2}$, γ с точностью 15 знаков после запятой. Для каждой константы реализовать два способа вычисления: например, как сумму ряда, как решение специального уравнения, как значение предела. <ol style="list-style-type: none"> a) Написать программу для обработки последовательности символов, разделенных пробелами. Необходимо вычислить их среднее арифметическое. Примечание. Ввод символа «не цифры» является корректным. Ввод данных осуществлять в зависимости от флагов, которые передаются, как аргументы командной строки: -s или отсутствие флага означает считывание данных с клавиатуры в интерактивном режиме; -f <file name> означает что символы нужно считать из файла с именем <filename>. b) Написать программу для обработки числовых данных. Из файла или с консоли прочитать произвольное число чисел и найти их среднее геометрическое, среднее гармоническое. При вводе недопустимого символа вывести информацию об ошибке. Примечание. Допускается ввод целых и вещественных чисел. Разделителем целой и дробной части является символ точки или запятой. Интерфейс приложения реализовать так же, как и в задаче a). 3. Пользователем задается строка или из файла, или с клавиатуры. <ol style="list-style-type: none"> a) Выполнить сортировку слов строки по алфавиту и вывести на экран слово, состоящее из последних символов этих слов. b) В каждом слове строки поднять регистр первой буквы слова и опустить регистр последней буквы. c) Подсчитать сколько раз в этой строке встречается заданное слово. d) Заменить в данной строке предпоследнее слово на слово, которое ввел пользователь. e) Найти k – ое слово в строке начинающиеся с заглавной буквы. 4. «FileViewer». На форме находится элемент управления TreeView. При выборе в меню пункта открыть появляется окно диалога открытия, в котором происходит выбор папки, после чего структура папки со всеми файлами появляется в элементе управления TreeView. При нажатии дважды на файл он открывается соответствующей программой или в отдельной форме показывается его 16-ое представление. При нажатии на файл или на папку правой кнопкой мыши появляется контекстное меню со следующими пунктами: Копировать, Переместить, Удалить, Переименовать, Создать, Посмотреть атрибуты, Наблюдать. Все действия реализовать посредством запроса модального диалога. Наблюдать за файлом значит вести лог-файл, в

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p>который будут заноситься все изменения, связанные с файлом; так же необходимо реализовать настройку изменений: запрет переименования, запрет модификации и т. д. Примечание. При обработке данных программа не должна «зависать».</p> <p>5. «Конструктор каталогов». Разработайте оконное приложение, позволяющее пользователю с помощью мыши создавать и редактировать структуру выбранного каталога. Строительными элементами являются каталоги (папки), которые могут иметь имена, а могут и не иметь их, и связи между ними, которые имеют направление. Данные строительные элементы располагаются на панели инструментов. Направление связи определяет тип каталога: дочерний или родительский. При этом у каталога может быть сколько угодно дочерних каталогов и только один родительский. Реализуйте возможность сохранения и восстановления построенной структуры, возможность создать реальную иерархию папок на компьютере с добавлением ее в указанное место и на основе имеющегося каталога построить его структуру. При реализации приложения используйте стандартный функционал сереализации. Так же создайте пользовательскую сереализацию: создайте класс, реализовав интерфейс ISerialization и добавив поддержку сереализации и десереализации. Сделайте процесс сереализации по возможности оптимальным. Обработайте все четыре события BinaryFormatter. Реализуйте интерфейс IFormatter в собственном форматирующем объекте. Задействуйте его в процессе сереализации и десереализации.</p> <p>6. «Построитель графиков». Разработать оконное приложение, которое по введенной формуле построит график данной функции. В окне ввода разрешается вводить формулы, которые содержат только одну неизвестную с именем x. При этом формула может быть произвольной: содержать скобки, символы известных констант, имена стандартных функций, другие константы и т.д. После подтверждения ввода в декартовой системе координат рисуется график функции. При этом необходимо предусмотреть возможности масштабирования, подписей осей координат и другие стандартные вещи. В одной системе координат может быть выведено разное количество графиков, соответственно необходимо менять текущее перо, которым выполняется рисование. В ваше приложение добавьте следующие опции: вывод координат точки графика, рядом с которой в данный момент находится курсор мыши; именование отмеченных точек; штриховку области под графиком функции между точками x_1 и x_2 (область, которую необходимо заштриховать, может быть не одна); проведение касательной или нормали в заданной точке; вычисление площади под графиком функции между точками x_1 и x_2. Так же в вашем приложении необходимо реализовать возможность настройки параметров вывода: цвет, толщина и тип текущего пера, вывод координатной сетки; и возможность сохранения в файл результатов работы вашей программы.</p>

5.2. Критерии, шкалы оценивания текущего контроля успеваемости:

Наименование оценочного средства (контрольно-оценочного мероприятия)	Критерии оценивания	Шкалы оценивания	
		100-балльная система	Пятибалльная система
Лабораторные работы	Обучающийся демонстрирует грамотное решение всех задач по программированию (программы работают и не выдают ошибок), использование правильных методов решения при незначительных погрешностях, не влияющих на работоспособность написанного кода.		5
	Продемонстрировано использование правильных методов написания программ при наличии 1-2 существенных ошибок.		4
	Обучающийся использует верные методы решения при написании программ, но выполненные верно задания в большинстве случаев отсутствуют.		3
	Обучающийся использует неверные методы решения при написании программ, большинство задач выполнены некорректно.		2

5.3. Промежуточная аттестация:

Форма промежуточной аттестации	Типовые контрольные задания и иные материалы для проведения промежуточной аттестации:
Зачет с оценкой (второй семестр): в устной форме по билетам	<p>Билет 1</p> <ol style="list-style-type: none"> Структурные и ссылочные типы. Правила приведения типов в C#. Разработать оконное приложение, в котором должны присутствовать кнопки, списки и переключатели. <p>Билет 2</p> <ol style="list-style-type: none"> Строковые типы, строковые потоки. Сравнение типов string, String и StringBuilder. Разработать оконное приложение, в котором должны присутствовать поле ввода, CheckBox и календарь. <p>Билет 3</p> <ol style="list-style-type: none"> Объектно-ориентированное программирование в C#. Инкапсуляция. Разработать оконное приложение, в котором должны присутствовать комбинированный список, NumericUpDown и панель.

Зачет (третий семестр): устный опрос по темам дисциплины	<ol style="list-style-type: none"> 1. Стандартные синтаксические конструкции языка. Условный оператор. Циклические операторы. 2. Классы и объекты. Правила определения классов, конструкторы, деконструкторы, инициализация объектов. Доступность классов, членов и типов. Ключевое слово this. Классы Environment и Console. 3. Статические методы, данные, классы. Модификаторы параметров методов. Аргументы по умолчанию. Расширяющие методы. 4. Структурные и ссылочные типы. Упаковка и распаковка. Отличие классов и структур. 5. Базовый класс object. Методы класса object. Стандартные типы данных. Типы, принимающие значение null. Особенности преобразования типов.
---	---

5.4. Критерии, шкалы оценивания промежуточной аттестации учебной дисциплины/модуля:

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
Наименование оценочного средства		100-балльная система	Пятибалльная система
зачет с оценкой: в устной форме по билетам	<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует знания, отличающиеся глубиной и содержательностью, дает полный исчерпывающий ответ, как на основные вопросы билета, так и на дополнительные; – свободно выполняет практические задания повышенной сложности, предусмотренные дисциплиной. Написанный программный код по выполнению практического задания не содержит ошибок и работает корректно. 		5
	<p>Обучающийся:</p> <ul style="list-style-type: none"> – показывает достаточное знание учебного материала, но допускает несущественные фактические ошибки, которые способен исправить самостоятельно, благодаря наводящему вопросу; – успешно выполняет предусмотренные в дисциплине практические задания средней сложности, написанный программный код по выполнению практического задания работает корректно, допущены лишь несущественные ошибки, которые исправимы в процессе обсуждения выполненного задания. 		4
	<p>Обучающийся:</p> <ul style="list-style-type: none"> – показывает знания фрагментарного характера, которые отличаются поверхностностью и малой содержательностью, допускает фактические грубые ошибки; – справляется с выполнением практических заданий, предусмотренных дисциплиной, знаком с основной литературой, рекомендованной программой, 		3

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
Наименование оценочного средства		100-балльная система	Пятибалльная система
	допускает погрешности и ошибки при теоретических ответах и в ходе практической работы при написании программ.		
	<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует существенные пробелы в знаниях основного учебного материала, допускает принципиальные ошибки в выполнении предусмотренных программой практических заданий; – на большую часть дополнительных вопросов по содержанию экзамена затрудняется дать ответ или не дает верных ответов. 		2
зачет: устный опрос по темам дисциплины	<p>Обучающийся:</p> <ul style="list-style-type: none"> – показывает достаточное знание учебного материала, но допускает несущественные фактические ошибки, которые способен исправить самостоятельно, благодаря наводящему вопросу; – успешно выполняет предусмотренные в дисциплине практические задания средней сложности, написанный программный код по выполнению практического задания работает корректно, допущены лишь несущественные ошибки, которые исправимы в процессе обсуждения выполненного задания. 		зачтено
	<p>Обучающийся:</p> <ul style="list-style-type: none"> – демонстрирует существенные пробелы в знаниях основного учебного материала, допускает принципиальные ошибки в выполнении предусмотренных программой практических заданий; – на большую часть дополнительных вопросов по содержанию экзамена затрудняется дать ответ или не дает верных ответов. 		не зачтено

5.5. Система оценивания результатов текущего контроля и промежуточной аттестации.

Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.

Форма контроля	100-балльная система	Пятибалльная система
Второй семестр		
Текущий контроль: Лабораторные работы		2 – 5
Промежуточная аттестация (зачет с оценкой)		отлично хорошо
Итого за семестр зачет с оценкой		удовлетворительно неудовлетворительно
Третий семестр		
Текущий контроль: Лабораторные работы		2 – 5
Промежуточная аттестация (зачет)		зачтено не зачтено
Итого за семестр зачет		

6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Реализация программы предусматривает использование в процессе обучения следующих образовательных технологий:

- поиск и обработка информации с использованием сети Интернет;
- дистанционные образовательные технологии в случае производственной необходимости;
- применение электронного обучения в случае производственной необходимости.

7. ПРАКТИЧЕСКАЯ ПОДГОТОВКА

Практическая подготовка в рамках учебной дисциплины реализуется при проведении лабораторных работ.

8. ОРГАНИЗАЦИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

При обучении лиц с ограниченными возможностями здоровья и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуального обучения, применение соответствующих методик по работе с инвалидами, использование средств дистанционного общения, проведение дополнительных индивидуальных консультаций по изучаемым теоретическим вопросам и практическим занятиям, оказание помощи при подготовке к промежуточной аттестации.

При необходимости рабочая программа дисциплины может быть адаптирована для обеспечения образовательного процесса лицам с ограниченными возможностями здоровья, в том числе для дистанционного обучения.

Учебные и контрольно-измерительные материалы представляются в формах, доступных для изучения студентами с особыми образовательными потребностями с учетом нозологических групп инвалидов.

Для подготовки к ответу на практическом занятии, студентам с ограниченными возможностями здоровья среднее время увеличивается по сравнению со средним временем подготовки обычного студента.

Для студентов с инвалидностью или с ограниченными возможностями здоровья форма проведения текущей и промежуточной аттестации устанавливается с учетом индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Промежуточная аттестация по дисциплине может проводиться в несколько этапов в форме рубежного контроля по завершению изучения отдельных тем дисциплины. При необходимости студенту предоставляется дополнительное время для подготовки ответа на зачете или экзамене.

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации обучающихся создаются, при необходимости, фонды оценочных средств, адаптированные для лиц с ограниченными возможностями здоровья и позволяющие оценить достижение ими запланированных в основной образовательной программе результатов обучения и уровень сформированности всех компетенций, заявленных в образовательной программе.

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Материально-техническое обеспечение дисциплины при обучении с использованием традиционных технологий обучения.

Наименование учебных аудиторий, лабораторий, мастерских, библиотек, спортзалов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.	Оснащенность учебных аудиторий, лабораторий, мастерских, библиотек, спортивных залов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.
119071, г. Москва, Малый Калужский переулок, дом 1, строение 2	
аудитории для проведения занятий лекционного типа	комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – ноутбук; – проектор; – проекционный экран.
аудитории для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – ноутбук; – проектор; – проекционный экран; – персональные компьютеры для обучающихся.
Помещения для самостоятельной работы обучающихся	Оснащенность помещений для самостоятельной работы обучающихся
читальный зал библиотеки	– компьютерная техника; подключение к сети Интернет.

10. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

№ п/п	Автор(ы)	Наименование издания	Вид издания (учебник, УП, МП и др.)	Издательство	Год издания	Адрес сайта ЭБС или электронного ресурса	Количество экземпляров в библиотеке Университета
10.1 Основная литература, в том числе электронные издания							
1	Гуриков С. Р.	Введение в программирование на языке Visual C#	Учебное пособие	М.: ФОРУМ: ИНФРА-М	2018	http://znanium.com/catalog/product/1017998	-
2	Хорев П. Б.	Объектно-ориентированное программирование с примерами на C#	Учебное пособие	М. : ФОРУМ : ИНФРА-М	2019	http://znanium.com/catalog/product/1018034	-
3	Белов В. В.	Алгоритмы и структуры данных	Учебник	М.: КУРС: ИНФРА-М	2020	https://znanium.com/catalog/product/1057212	-
10.2 Дополнительная литература, в том числе электронные издания							
1	Дадян Э. Г.	Современные технологии программирования. Язык C#: учебник: в 2 томах. Том 1. Для начинающих пользователей	Учебник	М: НИЦ ИНФРА-М	2021	https://znanium.com/catalog/product/1196552	-
2	Дадян Э. Г.	Современные технологии программирования. Язык C#: учебник: в 2 томах. Том 2. Для продвинутых пользователей	Учебник	М: НИЦ ИНФРА-М	2021	https://znanium.com/catalog/product/1478383	-
3	Царев Р. Ю.	Алгоритмы и структуры данных (CDIO)	Учебник	Краснояр.:СФУ	2016	https://znanium.com/catalog/product/967108	-

11. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

11.1. Ресурсы электронной библиотеки, информационно-справочные системы и профессиональные базы данных:

№ пп	Электронные учебные издания, электронные образовательные ресурсы
1.	ЭБС «Лань» http://www.e.lanbook.com/
2.	«Znanium.com» научно-издательского центра «Инфра-М» http://znanium.com/
3.	Электронные издания «РГУ им. А.Н. Косыгина» на платформе ЭБС «Znanium.com» http://znanium.com/
4.	Образовательная платформа «Юрайт» https://urait.ru/
5.	Электронные ресурсы «Polpred.com Обзор СМИ» https://www.polpred.com/
6.	Электронные ресурсы «Национальной электронной библиотеки» («НЭБ») https://rusneb.ru/
Профессиональные базы данных, информационные справочные системы	
1.	Информационно-аналитическая система SCIENCE INDEX (включенная в научный информационный ресурс eLIBRARY.RU) https://www.elibrary.ru/
2.	База данных Springer eBooks Collections издательства Springer Nature. Платформа Springer Link: https://rd.springer.com/
3.	Электронный ресурс Freedom Collection издательства Elsevier https://sciencedirect.com/
4.	База данных научного цитирования Scopus издательства Elsevier https://www.scopus.com/
5.	База данных ORBIT IPBI (Platinum Edition) компании Questel SAS https://www.orbit.com/
6.	База данных Web of Science компании Clarivate Analytics https://www.webofscience.com/wos/woscc/basic-search
7.	База данных CSD-Enterprise компании The Cambridge Crystallographic Data Center https://www.ccdc.cam.ac.uk/
8.	Научная электронная библиотека «elibrary.ru» https://www.elibrary.ru/
9.	База данных издательства SpringerNature https://link.springer.com/ https://www.springerprotocols.com/ https://materials.springer.com/ https://link.springer.com/search?facet-content-type=%ReferenceWork%22 http://zbmath.org/ http://npg.com/

11.2. Перечень программного обеспечения

№п/п	Программное обеспечение	Реквизиты подтверждающего документа/ Свободно распространяемое
1.	Windows 10 Pro, MS Office 2019	контракт № 18-ЭА-44-19 от 20.05.2019
2.	Microsoft Visual Studio	контракт № 18-ЭА-44-19 от 20.05.2019

ЛИСТ УЧЕТА ОБНОВЛЕНИЙ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

В рабочую программу учебной дисциплины внесены изменения/обновления и утверждены на заседании кафедры:

№ пп	год обновления РПД	характер изменений/обновлений с указанием раздела	номер протокола и дата заседания кафедры