

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ))



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы бакалавриата
по направлению подготовки
27.03.04 Управление в технических системах,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программирование и основы алгоритмизации

Направление подготовки: 27.03.04 Управление в технических системах

Направленность (профиль): Системы, методы и средства цифровизации и управления

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде
электронного документа выгружена из единой
корпоративной информационной системы управления
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 2053
Подписал: заведующий кафедрой Баранов Леонид Аврамович
Дата: 01.06.2024

1. Общие сведения о дисциплине (модуле).

Целями освоения учебной дисциплины (модуля) «Программирование и основы алгоритмизации» являются формирование у обучающихся системно-информационного взгляда на мир, включающего абстрагирование, моделирование и алгоритмическое мышление, обеспечение прочного овладения студентами основами знаний и практических навыков алгоритмизации задач и программирования на языках структурного и объектно-ориентированного программирования.

Основной задачей изучения учебной дисциплины «Программирование и основы алгоритмизации» является формирование у обучающегося компетенций для следующих типов задач профессиональной деятельности:

- проектно-конструкторской; - научно-исследовательской.

Дисциплина предназначена для получения знаний обучающимися для решения следующих профессиональных задач (в соответствии с указанными выше типами задач профессиональной деятельности):

проектно-конструкторская деятельность:

- формулирование целей проекта, критериев и способов достижения целей, построение структуры их взаимосвязей, выявление приоритетов решения задач;

- разработка обобщенных вариантов решения проблемы, их анализ, прогнозирование последствий, нахождение компромиссных решений в условиях многокритериальности и неопределенности, планирование реализации проекта;

- использование компьютерных технологий в проектно-конструкторской деятельности;

- проектирование решений, соответствующих современным достижениям науки и техники;

- разработка проектной и конструкторской документации для решения задач;

- разработка, согласование и подготовка к вводу в действие технических регламентов, других нормативных документов и руководящих материалов, связанных с проектированием, эксплуатацией и техническим обслуживанием решенных задач;

научно-исследовательская деятельность:

- сбор научной информации, подготовка обзоров, аннотаций, составление рефератов и отчетов, библиографий, анализ информации по объектам исследования;

- анализ и интерпретация на основе существующих научных концепций

отдельных явлений и процессов с формулировкой аргументированных умозаключений и выводов;

- проведение научных исследований в отдельных областях, связанных с организацией проектирования, историей науки и техники;

- участие в научных дискуссиях и процедурах защиты научных работ различного уровня;

- выступление с докладами и сообщениями по тематике проводимых исследований;

- анализ состояния и динамики объектов деятельности с использованием необходимых методов и средств анализа, моделирование исследуемых явлений или процессов с использованием современных вычислительных машин и систем, а также компьютерных программ;

- разработка программ и методик испытаний объектов, разработка предложений по внедрению результатов научных исследований.

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ОПК-5 - Способен решать задачи развития науки, техники и технологии в области управления в технических системах с учетом нормативно-правового регулирования в сфере интеллектуальной собственности;

ПК-1 - Способен принимать участие в разработке, исследовании эффективности функционирования и совершенствовании технических и программных средств автоматических и автоматизированных систем управления транспортными объектами;

ПК-4 - Способен проводить вычислительные эксперименты с использованием стандартных программных средств с целью получения математических моделей процессов и объектов автоматизации и управления.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Знать:

- современные информационные технологии, методы и средства контроля, диагностики и управления в своей профессиональной деятельности;

- технические и программные средства автоматических и автоматизированных систем управления транспортными объектами;

- стандартные программные средства с целью получения математических моделей процессов и объектов автоматизации и управления.

Уметь:

- использовать современные информационные технологии, типовые средства контроля, диагностики и управления технических систем.
- выбирать критерии и ставить задачи исследования эффективности функционирования и совершенствования технических и программных средств автоматических и автоматизированных систем управления транспортными объектами.
- применять современные программные и технические средства при разработке моделей АСУ, процессов и объектов автоматизации и управления.

Владеть:

- методиками исследования и повышения эффективности функционирования технических и программных средств автоматических и автоматизированных систем управления транспортными объектами
- выбирает метод ее решения с использованием современных, информационных технологий, и используемых в области управления, контроля и диагностики технических систем.

3. Объем дисциплины (модуля).**3.1. Общая трудоемкость дисциплины (модуля).**

Общая трудоемкость дисциплины (модуля) составляет 14 з.е. (504 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов			
	Всего	Семестр		
		№1	№2	№3
Контактная работа при проведении учебных занятий (всего):	224	64	96	64
В том числе:				
Занятия лекционного типа	112	32	48	32
Занятия семинарского типа	112	32	48	32

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 280 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	<p>Предмет программирования и основ алгоритмизации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятия, термины, предмет программирования; - основы алгоритмизации; - структура отчётной документации к решённым задачам алгоритмизации и программирования.
2	<p>Понятие алгоритма</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - основные понятия алгоритма; - базовые алгоритмические конструкции; - блок-схемы алгоритмов (нотация FlowChart Diagram); - запрещённые типы связей при изображении блок-схем алгоритмов; - Microsoft Office Visio как эффективный инструмент подготовки блок-схем алгоритмов.
3	<p>Программирование как планирование вычислительного процесса</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие выражений; - константы и переменные; - операнды и операторы; - арифметические выражения; - логические выражения; - приоритет выполняемых операций; - типы данных.
4	<p>Тестирование программного обеспечения</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - программа как «чёрный ящик»; - компиляторы, интерпретаторы, исполняемые файлы; - входная информация; - выходная информация; - подбор тестовых примеров; - точность производимых вычислений; - область значимости, область незначимости, эpsilon-окрестность точки; - области допустимых значений и определения функциональных зависимостей.
5	<p>Интегрированная среда разработки (IDE) Microsoft Visual Studio. Консольный режим работы</p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - начало работы с интегрированной средой разработки; - выбор настроек среды под создание шаблонов на языке Visual C#; - проверка наличия программируемых каркасов (.NET Framework); - выбор режимов разработки прикладного программного обеспечения; - командная строка ОС Microsoft Windows; - понятие программируемой консоли; - шаблон разработки на языке Visual C#.
6	<p>Базовые конструкции консольного режима работы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - раздел подключаемых модулей using; - системный программный модуль (using System); - пространства имён переменных (namespaces); - статические (static) конструкции языка; - класс «Программа» и главный метод программы.
7	<p>Последовательный вычислительный процесс. Типовые задачи</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - объявление переменных; - инициализация переменных; - организация небезопасного ввода данных (Convert); - организация безопасного ввода данных (Parse и TryParse); - преобразование типов данных (явное / неявное приведение типов данных); - оператор присвоения; - отладка программного обеспечения, точки останова; - оператор безусловного перехода (goto) для нужд отладки программного обеспечения; - типовая задача «Ввода-вывода»; - типовая задача «Ввода-решения-вывода»; - типовая задача «Три коробки» (обмен значениями переменных).
8	<p>Целостность прикладного программного обеспечения</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие решения (solution); - понятие проекта (project); - решение проблемы передачи проекта по электронной почте; - копирование проекта; - восстановление целостности проекта.
9	<p>Система контроля версий Git</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - установка системы контроля версий Git на персональный компьютер; - создание учётной записи на облачной платформе GitHub; - блокноты с разметкой синтаксиса; - настройка инструментов сравнения версий; - установка ГПИ-клиента Git Extensions, предназначенного для повышения эффективности работы с Git.
10	<p>Форматирование вывода результатов</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - вывод с конкатенацией строк; - параметрический вывод; - вывод с интерполяцией строк; - настройки консоли, обеспечивающие экономию расходных материалов при распечатке документации; - разметка областей составленного кода (#region... #endregion); - организация задержки консоли.

№ п/п	Тематика лекционных занятий / краткое содержание
11	<p>Правила вежливости программиста</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие о значимости имён переменных и программных конструкций; - нотации наименования переменных; - принцип однострочного размещения логически завершённых операций; - однострочные и многострочные комментарии; - документирование методов и полей; - правила объявления переменных; - сдвиги кодовых конструкций, определяющие вложенность операторов.
12	<p>Стандартные функции модуля «Математика» (Math)</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - статические методы; - перевод величин из градусов в радианы и из радианов в градусы; - одно- и многопараметрические функции; - соответствие типов данных переменных типам данных возвращаемых функцией значений; - прямой возврат значений и возврат изменённых значений входящих параметров.
13	<p>Разветвляющийся вычислительный процесс. Типовые задачи</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - разветвляющийся вычислительный процесс; - условный оператор как основной оператор разветвляющегося вычислительного процесса; - неполный условный оператор; - блоки истинного решения и ложного результата; - вложенный условный оператор; - унарные (один операнд) и бинарные (два операнда) операторы; - сокращённая форма условного оператора – тернарный (три операнда) оператор; - определение минимального и максимального значения из пары; - поиск идентичных среди введённых значений; - учёт уровня значимости (заданной точности), сопоставление в эпсилон-окрестностью точки; - использование значений отклика безопасного ввода данных TryParse для контроля достаточности исходной информации для выполнения расчётов; - флаги и ключи как инструменты хранения результатов расчёта сложных логических выражений; - расчёт систем уравнений, содержащих условия; - численная интерпретация геометрических задач.
14	<p>Оператор переключения. Типовые задачи</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - переменные переключения; - допустимые типы данных для переменных переключения; - особенности формирования оператора переключения с логической переменной переключения; - служебные слова и синтаксис оператора переключения; - особенности составления блок-схем, содержащих оператор переключения; - учёт нескольких меток для перехода к однотипным блокам.
15	<p>Перечислимый тип данных</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - формирование перечислимого типа данных; - работа с переменными перечислимого типа; - выдача непоследовательных идентификаторов перечислимым значениям; - стандартные перечислимые типы данных; - распознавание стандартных множеств, уместных для формирования перечисливых данных и и переменных перечислимого типа.
16	<p>Циклический, итерационный вычислительный процесс</p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - понятие о циклическом вычислительном процессе; - циклы с предусловием и с постусловием; - особенности иллюстрации итерационных циклов на блок-схемах алгоритмов; - понятие параметра цикла; - принудительное прерывание циклических конструкций; - служебное слово пропуска итерации.
17	<p>Циклический вычислительный процесс. Диапазоны. Типовые задачи</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие параметров диапазона; - операции инкрементирования и декрементирования; - числовой отрезок (счёт итераций с нуля и с единицы); - расчёт количества итераций через параметры диапазона; - особенности иллюстрации циклов по известным диапазонам значений на блок-схемах алгоритмов; - особенности передачи значений параметров цикла за пределы циклических конструкций; - табулирование значений аргументов и функций; - прохождение диапазона в прямом и обратном направлении; - прохождение диапазона с целым шагом; - пересчёт шага построения графика функциональной зависимости на основе целочисленного параметра цикла; - организация бесконечных циклов (итерационных петель); - типовая аддитивная задача (циклические сложения и вычитания); - типовая мультипликативная задача (циклическое умножение); - расчёт сумм знакопеременного ряда; - расчёт сумм знакопостоянного ряда.
18	<p>Одномерные массивы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие об одномерном массиве / векторе; - объявление одномерных массивов; - инициализация значений одномерных массивов; - базовые конструкции и алгоритмы работы с одномерными массивами; - задача упорядочивания значений в одномерном массиве; - увеличение / сокращение размерности одномерного массива.
19	<p>Многомерные массивы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - многомерные массивы; - объявление многомерных массивов; - базовые конструкции и алгоритмы работы с многомерными массивами; - схемы сортировки многомерных массивов; - послойный вывод многомерных массивов; - особые схемы вывода значений, хранимых в массивах, в окно консоли.
20	<p>Зубчатые массивы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие о зубчатых массивах; - примеры применения зубчатых массивов на практике; - базовые конструкции и алгоритмы работы с зубчатыми массивами.
21	<p>Строки и стандартные строковые методы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие строки как массива символов; - таблицы символов, работа с таблицами символов; - различие между строками и символами; - модуль-класс StringBuilder;

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - строковые директивы; - инструмент отключения директив в строках «@»; - строка как универсальный контейнер; - множество строковых методов.
22	<p>Регулярные выражения</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - типовые задачи разбора параметров сетевых ресурсов; - типовые задачи детектирования дата/временных параметров; - детекция библиографических ссылок в тексте и их автоматизированная корректировка.
23	<p>Типизированные методы (функции) и нетипизированные (пустые) методы – процедуры</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие о типизированных методах; - возвращаемые параметры, множество возвращаемых параметров; - модификаторы доступа для типизированных методов; - документирование типизированных методов; - свёртывание однотипных фрагментов кода; - понятие о нетипизированных (пустых) методах; - модификаторы доступа для нетипизированных методов; - документирование нетипизированных методов.
24	<p>Типовые задачи с рекурсией</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - мультипликативная задача – факториал; - аддитивные задачи – итерационное суммирование; - задачи комбинаторики.
25	<p>Лямбда-выражения. Анонимные методы. Делегирование.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие о лямбда-выражениях (анонимных методах); - примеры использования лямбда-выражений; - формирование методов-делегатов; - делегирование.
26	<p>Типовые способы сортировки значений</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - «пузырьковый» способ; - шейкерный способ; - расчёсочный способ; - вставочный способ; - быстрый способ.
27	<p>Введение в объектно-ориентированное программирование. Основные понятия объектно-ориентированного программирования</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - введение в объектно-ориентированное программирование; - термины объектно-ориентированного программирования; - примеры обеспечения удобства чтения программного кода; - абстракция: мир идей и мир вещей; - основные понятия объектно-ориентированного программирования; - понятие базового класса; - класс System.Object; - стандартный набор методов базового класса.
28	<p>Классы, объекты, структуры</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - классы, объекты, структуры; - значащие типы данных; - ссылочные типы данных; - понятие конструктора экземпляра класса; - модификаторы доступа для классов; - концепции модульного принципа: один файл – один класс.
29	<p>Отлов ошибок при работе со внешними источниками информации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие исключения (исключительной ситуации); - классификация исключений; - оператор отлова исключений <code>try...catch...finally</code>; - реакция инструмента отлова исключений на исключения различных классов.
30	<p>Файлы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - потоки чтения и потоки записи; - файлы; - работа с файлами; - понятие кодовых страниц; - настройка кодировки.
31	<p>Составляющие модуля «Ввода-вывода» (System.IO)</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - работа с файловой системой; - команды командной строки, транслируемые посредством языка C#; - проверка существования путей и каталогов; - работа с атрибутами файлов и каталогов.
32	<p>Режим разработки экранных форм среды программирования</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - знакомство с шаблоном режима проектирования экранных форм среды программирования; - графический конструктор экранных форм; - конструктор экземпляра класса экранной формы при рассмотрении экранной формы как объекта; - связь с консольным режимом разработки; - базовые интерфейсные элементы управления.
33	<p>События. Обработка событий</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - стандартные события интерфейсных элементов управления; - подключение второстепенных событий интерфейсных элементов управления; - понятие инициатора (<code>sender</code>) вызова события; - понятие о переменных и свойствах события.
34	<p>Дочерние экранные формы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - вызов дочерних экранных форм; - режимы закрытия дочерних экранных форм; - унификация обмена данными с дочерними экранными формами.
35	<p>Программная компоновка интерфейсных элементов управления</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятия абсолютного и относительного размещения; - ключевые принципы размещения элементов управления на экранной форме; - списки интерфейсных элементов управления; - масштабирование экранных форм;

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - соблюдение эргономических принципов проектирования экранных форм; - понятие минимальной экранной плотности; - связь позиций интерфейсных элементов управления с изменяющимся масштабом.
36	<p>Перегрузка операторов и методов</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - множество операторов предназначенных для перегрузки; - множество не перегружаемых операторов; - типовые задачи перегрузки операторов; - понятие сигнатуры методов; - допустимые возможности перегрузки; - перегрузка стандартных методов; - виртуальные методы.
37	<p>Структурные списки</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - особенности формирования списков; - статические списки; - списки экземпляров классов; - обобщённые списки.
38	<p>Циклические конструкции по известному множеству значений</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие множества значений; - рассмотрение списков как множеств; - использование служебного слова <code>in</code> для работы со множествами; - оператор цикла по известному множеству значений <code>foreach</code>.
39	<p>Текстовы форматы баз данных</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - обработка информации при чтении / записи значений, разделённых запятыми (<code>csv</code>); - обработка информации при чтении / записи объектов в объектной нотации JavaScript (<code>json</code>); - обработка информации при чтении / записи данных, размещённых внутри расширенной разметки (<code>xml</code>).
40	<p>Календари языка программирования</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - класс датавременных параметров <code>DateTime</code>; - интерфейсный элемент управления селектор времени и даты (<code>DateTimePicker</code>); - интерфейсный элемент управления календарь (<code>Calendar</code>); - работа с выбранным диапазоном дат; - маркировка значимых дат в календаре; - методы повышения интерактивности при работе с календарём.
41	<p>Инкапсуляция</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие о внутренних методах обработки экземплярами классов; - понятие о программируемых свойствах; - геттеры и сеттеры программируемых свойств; - формирование ограничений на ввод / чтение внутренних значений экземпляров классов.
42	<p>Работа с графикой посредством Visual C#</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - подключение модулей, предоставляющих возможность работы с графикой; - связь переменных обработки графики с интерфейсными элементами управления; - понятие о растровых изображениях; - получение информации об интенсивности цветовых каналов;

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - использование экранных форм в качестве холстов для рисования; - параметрическое хранение иллюстраций; - понятие о перерисовке; - события, связанные с перерисовкой графических объектов.
43	<p>Преобразование взаимно зависимых физических величин в экранные координаты</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие об экранных координатах; - построение координатных осей; - пересчёт координат относительно настраиваемых положений графических начал координат.
44	<p>Проектирование меню программных приложений</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие главного меню и настройка его позиций; - понятие выпадающего меню и настройка его позиций; - понятие контекстного меню и настройка его позиций; - сокрытие / отображение пунктов меню в зависимости от смены состояний программного приложения; - программная компоновка меню как основа его стабильной работы.
45	<p>Наследование</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие наследования классов; - типовые задачи, решаемые с использованием аппарата наследования классов; - роль перегрузок методов при наследовании; - обобщение при наследовании.
46	<p>Оператор-интерпретатор (as)</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие интерпретации в программировании; - интерпретирование экземпляров классов в обобщённых списках; - интерпретирование переменных и свойств, связанных с событиями.
47	<p>Интерфейсы</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие интерфейса в программном проекте; - формирование интерфейса; - типовые задачи, решаемые при помощи интерфейсов.
48	<p>Операции (инициализация, выполнение, отмена)</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие операции, как объекта; - схема инициализации операции; - планирование выполнения операции; - планирование отмены операции; - объединение операций интерфейсами; - списки выполненных и отменённых операций.
49	<p>Область для построения графиков и диаграмм (Chart)</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - размещение шаблона chart на экранной форме; - подключение модулей, повышающих эффективность взаимодействия с chart, к экранной форме; - настройка легенды, представления и оформления данных в chart; - передача структурных списков в chart.
50	<p>Подключаемые модули. Автоматизация электронного документооборота. Microsoft Office Word</p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - подключение модуля объектно-ориентированного взаимодействия с Word; - применение механизма отлова ошибок при автоматизации компоновки документов Word; - трассирование ошибок, возникающих в процессе компоновки; - соответствие между параметрами графического интерфейса Word и его объектной модели; - множества перечислимых типов, способствующих пониманию внутренних процессов Word.
51	<p>Разбор содержимого документов Microsoft Office Word</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - сценарий полной проходки документа; - применение механизма регулярных выражений для поиска интересующих фрагментов; - сценарий проходки табличных структур документа; - сценарий проходки абзацев документа; - понятие парсинга данных; - устранение служебных символов, заносимых в строки при парсинге данных Word.
52	<p>Подключаемые модули. Автоматизация электронного документооборота. Microsoft Office Excel</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - подключение модуля объектно-ориентированного взаимодействия с Excel; - понятия структуры таблиц при электронном документообороте; - метод эффективного обращения к ячейкам по номератором строк и столбцов; - автоматизированное формирование «чердака» и «подвала» таблицы; - автоматизированное формирование «шапки» таблицы; - автоматизированное формирование «тела» таблицы.
53	<p>Разбор содержимого электронных таблиц Microsoft Office Excel</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - идентификация рабочего диапазона; - получение данных из объединённых ячеек; - учёт оформления ячеек при переносе данных; - переключение между книгами и листами.
54	<p>Подключаемые модули. Автоматизация электронного документооборота. Microsoft Office Visio</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - подключение модуля объектно-ориентированного взаимодействия с Visio; - роль алгоритмической проработки при автоматизации составления документов Visio; - активация типовых шаблонов Visio; - использование стандартных фигур и наполнение их содержимым.
55	<p>Базовые сведения о потоках</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие многопоточности; - методы многопоточности; - типовые задачи, решаемые с применением многопоточности.
56	<p>Инструменты запросов, интегрированных в язык программирования (LINQ)</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - общее понятие о запросах, интегрированных в язык программирования; - методы расширения запросов, интегрированных в язык программирования; - предекаты / делегаты в запросах, интегрированных в язык программирования.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	<p>Формирование документации на разработку программного обеспечения</p> <p>В результате выполнения лабораторной работы студент отрабатывает умение по формированию документации на разработку программного обеспечения. Сценарий выполнения лабораторной работы включает в себя заготовку шаблона для последующего оформления решения задач алгоритмизации и программирования. Выполнение работы способствует запоминанию последовательности этапов разработки программного обеспечения, которая основывается на каскадной модели жизненного цикла программного обеспечения.</p>
2	<p>Схемы алгоритмов</p> <p>Сценарий выполнения лабораторной работы сводится к освоению студентом нотации составления схем алгоритмов (диаграмм потоков, FlowChart Diagram). Описание бытовой ситуации должно быть разобрано на элементарные шаги, из которых и составляется схема алгоритма. В результате выполнения работы студент отрабатывает умение по составлению схем алгоритмов для решения инженерных задач.</p>
3	<p>Типовая задачи «ввода-вывода» в консольном режиме разработки</p> <p>Сценарий выполнения лабораторной работы сводится к формированию текстовых сообщений в консоли по предложенным схемам: параметрического вывода, вывода с использованием механизма интерполяции строк, вывода по результатам конкатенации (слияния) строк, а также вывода в результате применения стандартных строковых методов. В результате выполнения работы студент отрабатывает умение по разработке вычислительных приложений в консольном режиме работы среды программирования с корректно оформленными побуждающими сообщениями для пользователя при вводе исходных данных и при выводе результата.</p>
4	<p>Типовая задача «ввода-решения-вывода» на примере решения сложного арифметического выражения</p> <p>Сценарий выполнения лабораторной работы включает преобразование из математической формы записи арифметического выражения в строковую форму записи, приемлемую для составления кода программы. По итогам выполнения лабораторной работы студент отрабатывает умение по разработке консольных приложений, содержащих расчёты в константах арифметических выражений, при которых важна явная типизация для получения корректных результатов (для констант различного типа одни и те же операторы работают по-разному – понимание перегрузки операторов в первом приближении).</p>
5	<p>Стандартные функции модуля «Математика»</p> <p>Сценарий выполнения лабораторной работы включает исследование студентом функционирования всех методов в составе статического класса «Math» из системной библиотеки классов («System»). По итогам выполнения лабораторной работы студент отрабатывает умение по разработке консольных приложений с использованием стандартных арифметических, тригонометрических, логарифмических и других математических функций.</p>
6	<p>Учёт прямых и косвенных ограничений при организации разветвления вычислительного процесса</p> <p>Сценарий выполнения лабораторной работы включает анализ студентом арифметических выражений, содержащих параметры, рассчитываемые в зависимости от выполнения или не выполнения указанных согласно индивидуальным вариантам условий, записанных явно логическими выражениями или неявно самой структурой построения арифметических выражений, а также тригонометрическими и логарифмическими функциями в их составе. В результате выполнения работы студент отрабатывает навыки разработки вычислительных приложений в консольном режиме работы среды программирования, содержащих разветвление вычислительного процесса.</p>
7	<p>Разветвление вычислительного процесса при решении задач в физической и геометрической постановках</p> <p>Сценарий выполнения лабораторной работы включает ознакомление студента с текстовыми постановками задач и принятие им решений о поиске корректных и непротиворечивых нормативно-справочных материалов, поставляющих данные и модели для решения геометрических и физических</p>

№ п/п	Наименование лабораторных работ / краткое содержание
	задач. В результате выполнения студент получает навык разработки программного обеспечения на основе исходных данных и расчётной модели, сформулированных текстом, составленным при минимуме математических записей и нормативно-справочных сведений.
8	<p>Консольные приложения для расчёта точек графиков функций</p> <p>Сценарий выполнения лабораторной работы включает построение циклического вычислительного процесса таким образом, чтобы расчёт не прерывался при попытках подстановки в качестве аргументов функций значений, расположенных вне области определения этих функций, предложенных согласно вариантам индивидуального задания. В результате выполнения студент отрабатывает умение по разработке консольных приложений для расчёта точек построения графиков функциональных зависимостей.</p>
9	<p>Планирование вычисления значений числовых рядов</p> <p>Сценарий выполнения лабораторной работы включает составление циклического вычислительного процесса, в теле которого на каждой итерации рассчитывается один из элементов либо знакопеременного, либо знакопостоянного числового ряда, который может сочетать в себе как чередующиеся тригонометрические функции, как значения, возводимые в степень, так и факториалы значений. В результате выполнения студент получает навык формализации элементов числовой последовательности.</p>
10	<p>Решение задач поиска экстремумов и детектирования значений, соответствующих заданным условиям</p> <p>Сценарий выполнения лабораторной работы включает составление циклического вычислительного процесса, в теле которого происходит непрерывный ввод значений. По завершении ввода значений необходимо в различных вариантах индивидуального задания давать ответы либо о найденном минимуме, либо о найденном максимуме, либо выдавать заключение относительно соответствия введённого значения установленным ограничениям (условиям). В результате выполнения лабораторной работы студент изучает диапазон значений и отрабатывает умение решать задачи поиска и детектирования.</p>
11	<p>Задачи упорядочивания, поиска и замены элементов в одномерных массивах</p> <p>Сценарий выполнения лабораторной работы включает формирование студентом одномерного массива (вектора) заданной размерности и заполнения его либо псевдослучайными значениями, либо вручную. Дальнейшие действия сводятся к формированию циклической конструкции, представляющей собой полный перебор элементов при фиксации индексов элементов, которые удовлетворяют установленным условиям по вариантам индивидуальных заданий – задача поиска. В отдельных вариантах поиск может сопровождаться обменом пары найденных значений – реализация задачи «Три коробки», а также необходимостью упорядочивания последовательности элементов – сортировка «Пузырьковым способом». Вывод массива должен быть реализован в математическом формате вида: «Name = (a1, ... aN)». В результате выполнения лабораторной работы студент отрабатывает умение решать задачи упорядочивания, поиска и замены элементов в одномерных массивах.</p>
12	<p>Задачи упорядочивания, поиска и замены в двумерных массивах</p> <p>Сценарий выполнения лабораторной работы включает формирование студентом двумерного массива (матрицы) заданных размерностей и заполнения его либо псевдослучайными значениями, либо вручную. Дальнейшие действия сводятся к формированию вложенных циклических конструкций, представляющих собой полный перебор элементов при фиксации индексов элементов, которые удовлетворяют установленным условиям по вариантам индивидуальных заданий – задача поиска. В отдельных вариантах поиск может сопровождаться обменом пары найденных значений – реализация задачи «Три коробки», а также необходимостью упорядочивания последовательности элементов – сортировка «Пузырьковым способом». Вывод массива должен быть реализован в математическом, табличном формате. В результате выполнения лабораторной работы студент отрабатывает умение решать задачи упорядочивания, поиска и замены элементов в двумерных массивах.</p>

№ п/п	Наименование лабораторных работ / краткое содержание
13	<p>Строки как универсальные контейнеры. Строковые методы</p> <p>Сценарий выполнения лабораторной работы включает работу с текстом – семантически значимой крупной строкой, содержащей более, чем одно предложение. Тексты, заданные по вариантам индивидуального задания, должны подвергаться автоматизированной правке и преобразованиям посредством использования стандартного набора методов в составе изучаемого языка программирования. Текст в индивидуальном задании может содержать и семантически не значимые конструкции, сопровождаемые идентичными разделителями. В такой ситуации содержимое необходимо последовательно извлекать из строки и выполнять его приведение к различным типам представления данных. В результате выполнения студент получает навык использования строковых методов при особом внимании к одному из них, реализующему разбиение содержимого на фрагменты, заключённые между фиксированными символами или наборами разделительных признаков.</p>
14	<p>Типизированные методы для составления программ с безызыбыточным кодом</p> <p>Сценарий выполнения лабораторной работы включает закрепление навыков работы с массивами и строками, которых к обработке в качестве индивидуального задания выдаётся более, чем в одном экземпляре. Требуется типовая обработка содержимого без повторяемости блоков кода. Соблюдение данного условия достигается посредством формирования студентом методов, представляющих собой поименованные фрагменты кода, к которым можно обращаться многократно с различными входными параметрами. В работе рассматриваются методы, которые в результате формируют значение или наборы значений, которые могут быть переданы на хранение в переменную соответствующего типа. В результате выполнения студент получает навык разработки типизированных методов для составления программных приложений с безызыбычным кодом.</p>
15	<p>«Пустые» методы для составления сценарных программ</p> <p>Сценарий выполнения лабораторной работы включает закрепление навыков работы с массивами и строками, которых к обработке в качестве индивидуального задания выдаётся более, чем в одном экземпляре. Требуется типовая обработка содержимого без повторяемости блоков кода. Соблюдение данного условия достигается посредством формирования студентом методов, представляющих собой поименованные фрагменты кода, к которым можно обращаться многократно с различными входными параметрами. В работе рассматриваются методы, представляющие собой сценарии. В результате выполнения студент получает навык разработки «пустых» методов для составления программных приложений с безызыбычным кодом.</p>
16	<p>Приложения для чтения и записи файлов с контролем наличия/отсутствия файлов</p> <p>Сценарий выполнения лабораторной работы включает ознакомление с инструментами библиотеки классов «Ввода-Вывода» (System.IO), поставляющей инструменты для работы с потоками данных на чтение (StreamReader) и на запись (StreamWriter). Поток данных (Stream), как правило, это сведения из внешнего источника, поведение которого может быть непредсказуемым, а потому методы, которые могут работать неустойчиво должны быть «обёрнуты» в работе либо собственным инструментарием контроля содержимого файловой системы, либо посредством оператора отлова исключений try ... catch ... finally. Вместе с тем работа позволяет отработать навык управления кодировкой (Encoding) текстового файла. В результате выполнения работы студент отработывает умение по созданию приложений для чтения и записи файлов с контролем наличия/отсутствия файлов.</p>
17	<p>Структуры данных</p> <p>Сценарий выполнения лабораторной работы включает составление унифицированных структур разнородной информации, которую необходимо укладывать в одну единственную переменную. В результате выполнения студент получает навык формирования структур данных, повышающих удобство хранения информации и удобочитаемость кода.</p>
18	<p>Данные перечислимого типа</p> <p>Сценарий выполнения лабораторной работы включает формирование переменных перечислимого типа для настройки соответствия между числовыми значениями и наименованиями, удобными для запоминания в ограниченных множествах, таких как: дни недели, месяцы, направления движения,</p>

№ п/п	Наименование лабораторных работ / краткое содержание
	клавиши на клавиатуре и т.п. В результате выполнения студент получает навык составлени удобочитаемых конструкций в коде программ.
19	Классы и объекты Сценарий выполнения лабораторной работы включает проработку класса под текстовое описание, выданное согласно варианту индивидуального задания, а также «упаковку» созданных экземпляров разработанного класса в список для последующей обработки и корректного вывода информации. Для корректного вывода информации переопределяется метод ToString(), относящийся к системному типу «Объект» (System.Object). В результате выполнения лабораторной работы студент отрабатывает умение по созданию классов, экземпляров классов и переопределению абстрактных методов в составе базового класса.
20	Инкапсуляция, перегрузка методов и операторов Сценарий выполнения лабораторной работы включает формирование класса для создания защищённых от прямых внешних воздействий объектов. Поля в таких классах делаются локальными по доступности, а изменение значений, хранимых в этих полях, прописывается через свойства, которые в особых случаях доступны на одновременные чтение и запись, а в большинстве известных ситуаций либо только доступны для чтения, либо только доступны для записи. В результате выполнения лабораторной работы студент приобретает навыки инкапсулирования классов, а также перегрузки методов и операторов, применяемых и относящихся к экземплярам таких классов.
21	Абстрактные классы. Наследование Сценарий выполнения лабораторной работы включает составление некоторого обобщающего, абстрактного класса (для которого не могут и не должны создаваться объекты), содержащего общие поля для всех наследников, а также составления классов-наследников (для которых уже могут и должны создаваться объекты), уточняющих специфику рассмотренного абстракта в рамках конкретной предметной области. В результате выполнения лабораторной работы студент приобретает навыки составления классов-наследников, основывающихся на общем абстрактном, базовом классе.
22	Обобщения. Полиморфизм Сценарий выполнения лабораторной работы включает составление студентом нескольких обобщённых методов, функционирующих одинаково для любых базовых типов данных, а также для любых классов, в какой бы момент жизненного цикла разработки программного обеспечения они ни появились. В результате выполнения лабораторной работы студент должен понимать практическую значимость такой составляющей объектно-ориентированного программирования как полиморфизм.
23	Интерфейсы Сценарий выполнения лабораторной работы включает формирование интерфейса, объединяющего классы со схожими методами. Дальнейшая работа сводится к «упаковке» экземпляров схожих классов в обобщённый интерфейс список и обработке каждого элемента этого списка в общем виде с вызовом соответствующих методов без преобразования типов объектов. В результате выполнения студент получает навык формирования связок, обобщающих классы по типовому набору входящих в их состав типизированных и нетипизированных методов.
24	Разработка приложений с графическим пользовательским интерфейсом Сценарий выполнения лабораторной работы включает знакомство студента с режимом построения программного обеспечения с экранными формами операционной системы Microsoft Windows. Работа посвящена ознакомлению с собственными свойствами экранных форм и событиями в составе одиночной экранной формы. В результате выполнения работы студент получает навык разработки графического пользовательского интерфейса для многофункционального ПО-вычислителя.
25	Модели систем, управляемых по событиям Сценарий выполнения лабораторной работы включает создание графической модели и текстового описания к ней технологической карты реализации бытовой ситуации из числа предложенных по индивидуальным вариантам заданий. Технологический процесс, протекающий во времени, но изменяющий состояние только в моменты управления оператором, рекомендовано рассматривать при

№ п/п	Наименование лабораторных работ / краткое содержание
	использовании аппарата сетей Петри, в котором время до изменения состояния не берется в расчёт, а важны только факты свершения некоторых событий. Эта нотация прекрасно подходит для описания систем, управляемых по событиям, к числу которых относятся программные системы с графическим пользовательским интерфейсом. В результате выполнения студент получает навык составления моделей технологических процессов работы программного обеспечения в нотации сетей Петри.
26	Дочерние экранные формы и статические классы Сценарий выполнения лабораторной работы включает создание в графическом конструкторе интегрированной среды разработки Microsoft Visual Studio нескольких экранных форм, а также увязку их между собой посредством активации обработчиков событий интерфейсных элементов управления, главным образом, – экранных кнопок, размещаемых на экранных формах. Экранные формы должны беспрепятственно обмениваться общедоступными для них данными, для чего студентом создаётся, структурно прорабатывается и размещается промежуточный статический класс-модуль «Данные». В результате выполнения студент получает навык разработки программного обеспечения с более, чем одной экранной формой.
27	Формирование графического пользовательского интерфейса кодом Сценарий выполнения лабораторной работы включает макетирование студентом технологической карты функционирования приложения, включающей все размещаемые на главной и дочерних экранных формах интерфейсные элементы управления, а также связи главной и дочерних экранных форм между собой через ключевые интерфейсные элементы управления, главным образом, экранные кнопки. Разрабатываемое приложение должно снабжаться справочной подсистемой, подгружающей информацию из текстовых файлов. В результате выполнения студент получает навык разбора и модифицирования содержимого автоматически собираемого раздела графического конструктора экранных форм.
28	Элементы иллюстрирования и программной анимации Сценарий выполнения лабораторной работы сводится к программному иллюстрированию на различных интерфейсных элементах управления объектов с возможностью их перемещения, удаления, восстановления и взаимного размещения. В результате выполнения студент получает навык работы с инструментами программного рисования в интегрированной среде разработки Microsoft Visual Studio, а также понимает событийные связи, обуславливающие необходимость перерисовки программных изображений по мере внесения изменений в состав графических объектов, а также в параметры элементов, поверх которых размещаются графические объекты.
29	Автоматизация записи данных в электронных документах Microsoft Office Word Сценарий выполнения лабораторной работы сводится к автоматизированному воспроизведению в текстовых документах Microsoft Office Word структур, предложенных согласно вариантам индивидуального задания в виде иллюстраций. В результате выполнения студент получает навык работы с программными инструментами, входящими в состав динамической библиотеки и реализующими объектно-ориентированный подход к программированию офисного приложения Microsoft Office Word.
30	Автоматизация записи данных в электронных таблицах Microsoft Office Excel Сценарий выполнения лабораторной работы сводится к автоматизированному воспроизведению в электронных таблицах Microsoft Office Excel таблиц, предложенных согласно вариантам индивидуального задания в виде иллюстраций. В результате выполнения студент получает навык работы с программными инструментами, входящими в состав динамической библиотеки и реализующими объектно-ориентированный подход к программированию офисного приложения Microsoft Office Excel.
31	Автоматизация составления схем в векторном редакторе Microsoft Office Visio Сценарий выполнения лабораторной работы сводится к автоматизированному воспроизведению в векторном редакторе Microsoft Office Visio иерархических схем или таблиц, предложенных согласно вариантам индивидуального задания в виде иллюстраций. В результате выполнения студент получает навык работы с программными инструментами, входящими в состав динамической библиотеки и

№ п/п	Наименование лабораторных работ / краткое содержание
	реализующими объектно-ориентированный подход к программированию офисного приложения Microsoft Office Visio.
32	Формирование запросов, интегрированных в язык программирования Сценарий выполнения лабораторной работы сводится к анализу выданного по варианту индивидуального задания описания некоторой предметной области, формированию подходящих значений параметров экземпляров, явно определяемых из описания классов, а также настройки фильтрации подмножеств внутри созданных множеств данных. В результате выполнения студент получает навык эффективного использования запросов, встроенных в синтаксис языка C#. Навык, в свою очередь, закладывает основы для эффективного и более быстрого освоения баз данных и систем управления базами данных.
33	Построение графиков и диаграмм с использованием Chart-области Сценарий выполнения лабораторной работы сводится к анализу выданной иллюстрации графика или диаграммы (в зависимости от варианта индивидуального задания) и воспроизведению её при помощи Chart-области. Студенту важно правильно сформировать перечни данных, пригодные для подачи на вход Chart-области, а также настроить графический пользовательский интерфейс приложения таким образом, чтобы предоставить пользователю возможность управления способами отображения графиков или диаграмм, а также координатными осями и сеткой. В результате выполнения студент получает навык использования и настройки компонента интегрированной среды разработки Microsoft Visual Studio, предназначенного для удобного построения графиков и диаграмм.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение дополнительной литературы.
2	Подготовка к лабораторным работам.
3	Выполнение курсовой работы.
4	Подготовка к промежуточной аттестации.
5	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых работ

Разработка учебных вычислительных приложений в консольном режиме работы среды программирования – backend-разработка.

Разработка консольных приложений для табулированного расчёта точек, необходимых для построения графиков функциональных зависимостей.

Составление детализированных и укрупнённых алгоритмов бытовых ситуаций, инженерных расчётов и технологических процессов.

Разработка графического пользовательского интерфейса многофункционального вычислителя – frontend-разработка.

Разработка приложений для чтения и записи файлов с контролем наличия/отсутствия файлов. Моделирование взаимодействия с базой данных на текстовых форматах xml, json, csv.

Разработка приложения, демонстрирующего построение графиков функциональных зависимостей с предварительным разбором строки, содержащей совокупность переменных, операций, тригонометрических и логарифмических функций и констант.

Разработка простого графического редактора, предоставляющего возможность для сохранения макета изображения файл и выгрузки результирующего изображения в одном из predetermined форматов (png, bmp, jpg, gif).

Разработка приложения для формирования сводного расписания преподавателей кафедры, а также индивидуальных карточек расписаний преподавателей по информации, размещённой в файле формата csv.

Разложение поступающих электронных документов формата Microsoft Office Word на типовые составляющие – парсеры Word.

Разработка упрощённого программного аналога электронных таблиц Microsoft Office Excel.

Разработка поисковой машины для учёта и индексации публикаций, необходимых для выполнения работ по выбранной тематике Проектной Деятельности.

Разработка программы, напоминающей о важных для пользователя событиях.

Разработка программы для протокольного обмена данными между двумя взаимодействующими системами.

Составление программного модуля для формирования шаблонных иллюстраций и вставки шаблонных иллюстраций на холст по указанной точке привязки.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Лабораторный практикум «Основы визуального программирования на языке С#»: Учебное пособие / Л. В. Дружинина, И. В. Приходькова.	https://elibrary.ru/item.asp?id=65202367

	<p>Дружинина, Л. В. Волгоград : Волгоградский государственный технический университет. – 64с. – ISBN 978- 5-9948-4859-3. – EDN LBRJWP. , 2024</p>	
2	<p>Составление отчётной документации по решённым задачам алгоритмизации и программирован ия : Учебно- методическое пособие для проведения аудиторных занятий по учебной практике А.И. Сафронов, Н.Н. Зольникова, В.Г. Новиков Книга Москва : Российский университет транспорта, 2018 – 83 с. – EDN WNAIFP. , 2018</p>	<p>https://elibrary.ru/item.asp?id=46271697</p>
3	<p>Получение первичных профессиональн ых умений и навыков научно- исследовательско й деятельности : Сборник задач для проведения аудиторных занятий по</p>	<p>https://elibrary.ru/item.asp?id=46307421</p>

	<p>учебной практике / А.И. Сафронов, Н.Н. Зольникова, В.Г. Новиков. Сафронов А.И. Москва: Российский университет транспорта, – 91 с. – EDN SXMWOD. , 2019</p>	
4	<p>Программирование на языке С# : учебное пособие / Р.Г. Гильванов, Л.М. Божко, А.Д. Хомоненко, И.Д. Липанов. Санкт- Петербург: ПГУПС, – 89 с. – ISBN 978-5-7641- 1977-9. , 2024</p>	<p>https://elibrary.ru/item.asp?id=69173205</p>
5	<p>Программирование на языке С#: практикум : учебное пособие А. А. Бармина, К. В.Вальштейн. Бармина А.А. Санкт-Петербург : БГТУ "Военмех" им. Д.Ф. Устинова, – 91 с. , 2022</p>	<p>https://reader.lanbook.com/book/382178?lms=40b1efc22d6eaae2e70118494a3f694c</p>
6	<p>Программирование на языке С# в среде Microsoft Visual Studio : учебно- методическое пособие / И. И. Данилина. Данилина, И.И. Екатеринбург: – 65 с. , 2018</p>	<p>https://reader.lanbook.com/book/121392?lms=61e7f167db6cd213f42563b2293b9036</p>

7	<p>Разработка прикладных программ в Microsoft Visual Studio C# : Лабораторный практикум / В. А. Комочков, А. А. Бурханов ; Комочков В.А. Волгоград: Волгоградский государственный технический университет. - 76 с. – ISBN 978-5-9948-3295-0. – EDN YIYZOX. , 2019</p>	<p>https://elibrary.ru/item.asp?id=37823182</p>
---	---	--

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Официальный сайт РУТ (МИИТ) (<https://www.miit.ru/>).

Научно-техническая библиотека РУТ (МИИТ) (<http://library.miit.ru>).

Образовательная платформа «Юрайт» (<https://urait.ru/>).

Общие информационные, справочные и поисковые системы «Консультант Плюс», «Гарант».

Электронно-библиотечная система издательства «Лань» (<http://e.lanbook.com/>).

Электронно-библиотечная система ibooks.ru (<http://ibooks.ru/>).

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Microsoft Internet Explorer (или другой браузер).

Операционная система Microsoft Windows.

Microsoft Office.

Microsoft Visual Studio 2015, Microsoft Visio.

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Учебные аудитории для проведения учебных занятий, оснащенные компьютерной техникой и наборами демонстрационного оборудования.

9. Форма промежуточной аттестации:

Экзамен в 1 семестре.

Зачет во 2, 3 семестрах.

Курсовая работа в 3 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

доцент, доцент, к.н. кафедры
«Управление и защита информации»

А.И. Сафронов

Согласовано:

Заведующий кафедрой УиЗИ
Председатель учебно-методической
комиссии

Л.А. Баранов

С.В. Володин