

Документ подписан простой электронной подписью
Информация о владельце: МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФИО: Силин Яков Петрович
Должность: Ректор
Дата подписания: 08.06.2026 15:31:34
Уникальный программный ключ:
24f866be2aca16484036a8cbb5c509a9551e605f

ФГБОУ ВО «Уральский государственный экономический университет»

09.12.2025 г.
протокол № 12
И.о. зав. кафедрой Кольева Н.С.

Утверждена
Советом по учебно-методическим
вопросам и качеству образования

16 декабря 2025 г.

протокол № 4
Председатель  Карх Д.А.



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины	Программирование
Направление подготовки	38.03.05 Бизнес-информатика
Профиль	Цифровой бизнес
Форма обучения	очная
Год набора	2026

Разработана:
Доцент, к.п.н.
Кольева Н.С.

Ст. преподаватель
Панова М.В.

Екатеринбург
2025 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ	3
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП	3
3. ОБЪЕМ ДИСЦИПЛИНЫ	3
4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ОПОП	3
5. ТЕМАТИЧЕСКИЙ ПЛАН	4
6. ФОРМЫ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ШКАЛЫ ОЦЕНИВАНИЯ	5
7. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	6
8. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ	9
9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ	9
10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ОНЛАЙН КУРСОВ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ	10
11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ	11

ВВЕДЕНИЕ

Рабочая программа дисциплины является частью основной профессиональной образовательной программы высшего образования - программы бакалавриата, разработанной в соответствии с ФГОС ВО

ФГОС ВО	Федеральный государственный образовательный стандарт высшего образования- бакалавриат по направлению подготовки 38.03.05 Бизнес-информатика (приказ Минобрнауки России от 29.07.2020 г. № 838)
---------	--

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Формирование первоначальных знаний, умений и навыков разработки программных средств на языках высокого уровня.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к обязательной части учебного плана.

3. ОБЪЕМ ДИСЦИПЛИНЫ

Промежуточная аттестация	Часов					З.е.
	Всего за семестр	Контактная работа (поуч.зан.)			Самостоятельная работа в том числе подготовка контрольных и курсовых	
		Всего	Лекции	Лабораторные		
Семестр 2						
Экзамен	144	48	16	32	69	4

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ОПОП

В результате освоения ОПОП у выпускника должны быть сформированы компетенции, установленные в соответствии с ФГОС ВО.

Шифр и наименование компетенции	Индикаторы достижения компетенций
ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации;	ИД-1.ОПК-3 Знать: принципы и методы организации управления процессами создания и использования продуктов и услуг в сфере ИКТ

<p>ОПК-3 Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации;</p>	<p>ИД-2.ОПК-3 Уметь: организовывать и управлять процессами создания и использования информационных продуктов и услуг</p>
	<p>ИД-3.ОПК-3 Иметь практический опыт: решения конкретных задач разработки, создания и использования программных продуктов.</p>
<p>ОПК-4 Способен понимать принципы работы информационных технологий; использовать информацию, методы и программные средства ее сбора, обработки и анализа для информационно-аналитической поддержки принятия управленческих решений;</p>	<p>ИД-1.ОПК-4 Знать: основные методы и подходы к поиску, сбору, обработке, анализу и систематизации информации, использованию программных средств и глобальных компьютерных сетей для подготовки обзоров, отчетов и научных публикаций</p>
	<p>ИД-2.ОПК-4 Уметь: использовать информационно-коммуникационные технологии, информационные ресурсы в решении профессиональных задач; самостоятельно проводить анализ информации, делать обоснованные выводы</p>
	<p>ИД-3.ОПК-4 Иметь практический опыт: применения методов системного анализа; инструментов математического моделирования, владения навыками использования программных продуктов для реализации типовых процедур обработки информации, методами анализа данных различного характера</p>

5. ТЕМАТИЧЕСКИЙ ПЛАН

Тема	Часов
------	-------

	Наименование темы	Всего часов	Контактная работа (по уч.зан.)			Самост. работа	Контроль самостоятельной работы
			Лекции	Лабораторные	Практические занятия		
Семестр 2		11					
Тема 1.	Введение в язык программирования	18	2	2		14	
Тема 2.	Компиляция и выполнение	14	2	2		10	
Тема 3.	Управляющие операторы (условия, циклы) (ОПК-3,	14	2	4		8	
Тема	Массивы (ОПК-3, ОПК-	14	2	4		8	
Тема 5.	Списки, строки и файлы (ОПК-3, ОПК-4)	14	2	4		8	
Тема	Основы тестирования	13	2	4		7	
Тема	Рекурсия (ОПК-3, ОПК-	15	2	6		7	
Тема	Поиск и сортировка	15	2	6		7	

6. ФОРМЫ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ШКАЛЫ ОЦЕНИВАНИЯ

Раздел/Тема	Вид оценочного	Описание оценочного средства	Критерии оценивания
Текущий контроль (Приложение 4)			
Темы 1-3	Тест (приложение)	Тест состоит из 31 вопроса	10 баллов
Темы 4-6	Контрольная работа (приложение)	Задания на программирование	10 баллов
Темы 7-8	Контрольная работа (приложение)	Задания на программирование	10 баллов
Промежуточная аттестация (Приложение 5)			
2 семестр (Эк)	Экзаменационный билет (приложение)	2 теоретических вопроса, 1 практическое задание	100 баллов

ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

Показатель оценки освоения ОПОП формируется на основе объединения текущего контроля и промежуточной аттестации обучающегося.

Показатель рейтинга по каждой дисциплине выражается в процентах, который показывает уровень подготовки студента.

Текущий контроль. Используется 100-балльная система оценивания. Оценка работы студента в течение семестра осуществляется преподавателем в соответствии с разработанной им системой оценки учебных достижений в процессе обучения по данной дисциплине.

В рабочих программах дисциплин и практик закреплены виды текущего контроля, планируемые результаты контрольных мероприятий и критерии оценки учебных достижений.

В течение семестра преподавателем проводится не менее 3-х контрольных мероприятий, по оценке деятельности студента. Если посещения занятий по дисциплине включены в рейтинг, то данный показатель составляет не более 20% от максимального количества баллов по дисциплине.

Промежуточная аттестация. Используется 5-балльная система оценивания. Оценка работы студента по окончании дисциплины (части дисциплины) осуществляется преподавателем в соответствии с разработанной им системой оценки достижений студента в процессе обучения по данной дисциплине. Промежуточная аттестация также проводится по окончании формирования компетенций.

Порядок перевода рейтинга, предусмотренных системой оценивания, по дисциплине, в пятибалльную систему.

Высокий уровень – 100% - 70% - отлично, хорошо.

Средний уровень – 69% - 50% - удовлетворительно.

Показатель оценки	По 5-балльной системе	Характеристика показателя
100% - 85%	отлично	обладают теоретическими знаниями в полном объеме, понимают, самостоятельно умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов на высоком уровне
84% - 70%	хорошо	обладают теоретическими знаниями в полном объеме, понимают, самостоятельно умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов. Могут быть допущены недочеты, исправленные студентом самостоятельно в процессе работы (ответаи т.д.)
69% - 50%	удовлетворительно	обладают общими теоретическими знаниями, умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов на среднем уровне. Допускаются ошибки, которые студент затрудняется исправить самостоятельно.
49 % и менее	неудовлетворительно	обладают не полным объемом общих теоретическими знаниями, не умеют самостоятельно применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов. Не сформированы умения и навыки для
100% - 50%	зачтено	характеристика показателя соответствует «отлично», «хорошо», «удовлетворительно»
49 % и менее	не зачтено	характеристика показателя соответствует «неудовлетворительно»

7. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

7.1. Содержание лекций

<p>Тема 1. Введение в язык программирования высокого уровня (ОПК-3, ОПК-4) Строки. Арифметические операции и var. Методы переменные.</p>
<p>Тема 2. Компиляция и выполнение программы (ОПК-3, ОПК-4) Стилистические ошибки. Рефакторинг кода. Константы и enum'ы. Выделение методов</p>
<p>Тема 3. Управляющие операторы (условия, циклы) (ОПК-3, ОПК-4) Цикл While. Цикл For. Сравнение циклов.</p>
<p>Тема 4. Массивы (ОПК-3, ОПК-4) Типы ссылки и типы значения. Передача массива в метод. Многомерные массивы. Массивы массивов.</p>
<p>Тема 5. Списки, строки и файлы (ОПК-3, ОПК-4) Специальные символы и форматированный вывод. Файлы и каталоги. Кодировка и работа с файлами.</p>
<p>Тема 6. Основы тестирования (ОПК-3, ОПК-4) Покрывание тестами. Функциональное тестирование.</p>
<p>Тема 7. Рекурсия (ОПК-3, ОПК-4) Рекурсия. Дерево рекурсии. Принцип "разделяй и властвуй". Перебор подмножеств. Перестановки. Размещения.</p>
<p>Тема 8. Поиск и сортировка (ОПК-3, ОПК-4) Сортировка пузырьком. Сортировка слиянием. Быстрая сортировка. Сравнение алгоритмов сортировки.</p>

7.2 Содержание практических занятий и лабораторных работ

<p>Тема 2. Компиляция и выполнение программы (ОПК-3, ОПК-4) Ошибки на этапе компиляции. Ошибки на этапе выполнения. Отладка.</p>
<p>Тема 3. Управляющие операторы (условия, циклы) (ОПК-3, ОПК-4) Операторы сравнения и логический тип данных. Сравнение чисел с плавающей точкой. Полные исокращенные операции. Операторы If и Else. Оператор Switch - Case.</p>
<p>Тема 4. Массивы (ОПК-3, ОПК-4) Массивы и цикл foreach. Короткая форма записи.</p>
<p>Тема 5. Списки, строки и файлы (ОПК-3, ОПК-4) Списки. Словари. Сравнение строк и массивов. Класс StringBuilder.</p>

<p>Тема 6. Основы тестирования (ОПК-3, ОПК-4)</p> <p>Введение в тестирование. Модульное тестирование.</p>
<p>Тема 7. Рекурсия (ОПК-3, ОПК-4)</p> <p>Сложность алгоритмов. Масштаб роста функций. O-символика. Оценка сложности. Сложность алгоритмов с числами. Классы сложности.</p>
<p>Тема 8. Поиск и сортировка (ОПК-3, ОПК-4)</p> <p>Линейный и бинарный поиск. Анализ линейного и бинарного поиска. Сравнение производительности. Создание графиков.</p>

7.3. Содержание самостоятельной работы

<p>Тема 2. Компиляция и выполнение программы (ОПК-3, ОПК-4)</p> <p>Решение задач "Angry Birds", "Бильярд", "Проценты".</p>
<p>Тема 3. Управляющие операторы (условия, циклы) (ОПК-3, ОПК-4)</p> <p>Решение задач "Рубли", "Два прямоугольника", "Расстояние до отрезка".</p> <p>Работа над проектом по построению путей выхода из лабиринта.</p>
<p>Тема 4. Массивы (ОПК-3, ОПК-4)</p> <p>Работа над проектом по построению гистограмм и тепловой карты.</p>
<p>Тема 5. Списки, строки и файлы (ОПК-3, ОПК-4)</p> <p>Работа над проектом по анализу текстов.</p>
<p>Тема 6. Основы тестирования (ОПК-3, ОПК-4)</p> <p>Работа над проектом по созданию парсера полей. Решение задач "Оттенки серого", "Медианный фильтр", "Пороговый фильтр", "Фильтр Собеля".</p>
<p>Тема 7. Рекурсия (ОПК-3, ОПК-4)</p> <p>Решение задач "Перебор паролей 2", "Хождение по чекпоинтам".</p>
<p>Тема 8. Поиск и сортировка (ОПК-3, ОПК-4)</p> <p>Решение задач "Левая граница", "Правая граница", "Автодополнение".</p> <p>Решение задач "Манипулятор", "Визуализация", "Поиск угла", "Решение манипулятора".</p>

7.3.1. Примерные вопросы для самостоятельной подготовки к зачету/экзамену
Приложение 1.

7.3.2. Практические задания по дисциплине для самостоятельной подготовки к зачету/экзамену
Приложение 2.

7.3.3. Перечень курсовых работ
Не предусмотрены.

7.4. Электронное портфолио обучающегося
Материалы не размещаются.

7.5. Методические рекомендации по выполнению контрольной работы
Не предусмотрены.

7.6 Методические рекомендации по выполнению курсовой работы
Не предусмотрены.

8. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

По заявлению студента

В целях доступности освоения программы для лиц с ограниченными возможностями здоровья при необходимости кафедра обеспечивает следующие условия:

- особый порядок освоения дисциплины, с учетом состояния их здоровья;
- электронные образовательные ресурсы по дисциплине в формах, адаптированных к ограничениям их здоровья;
- изучение дисциплины по индивидуальному учебному плану (вне зависимости от формы обучения);
- электронное обучение и дистанционные образовательные технологии, которые предусматривают возможности приема-передачи информации в доступных для них формах.
- доступ (удаленный доступ), к современным профессиональным базам данных и информационным справочным системам, состав которых определен РПД.

9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Сайт библиотеки УрГЭУ

<http://lib.usue.ru/>

Основная литература:

2. Корнеев В.И., Гагарина Л.Г., Корнеева М.В. Программирование графики на C++. Теория и примеры [Электронный ресурс]: Учебное пособие. - Москва: ООО "Научно-издательский центр ИНФРА-М", 2024. - 517 – Режим доступа: <https://znanium.com/catalog/product/2111934>

3. Гниденко И. Г., Павлов Ф. Ф., Федоров Д. Ю. Технологии и методы программирования [Электронный ресурс]: учебное пособие для вузов. - Москва: Юрайт, 2024. - 248 – Режим доступа: <https://urait.ru/bcode/536903>

4. Швецкий М. В., Демидов М. В., Голанова А. В., Кудрявцева И. Программирование: математическая логика [Электронный ресурс]: учебное пособие для вузов. - Москва: Юрайт, 2024. - 675 – Режим доступа: <https://urait.ru/bcode/542164>

5. Кудрина Е. В., Огнева М. В. Основы алгоритмизации и программирования на языке С# [Электронный ресурс]: учебное пособие для вузов. - Москва: Юрайт, 2024. - 322 – Режим доступа: <https://urait.ru/bcode/541687>

6. Трофимов В. В., Павловская Т. А. Алгоритмизация и программирование [Электронный ресурс]: учебник для вузов. - Москва: Юрайт, 2025. - 108 – Режим доступа: <https://urait.ru/bcode/562040>

7. Палий И. А. Линейное программирование [Электронный ресурс]: учебник для вузов. - Москва: Юрайт, 2025. - 175 – Режим доступа: <https://urait.ru/bcode/563472>

Дополнительная литература:

2. Гуриков С. Р. Введение в программирование на языке Visual C# [Электронный ресурс]: Учебное пособие. - Москва: Издательство "ФОРУМ", 2020. - 447 – Режим доступа: <https://znanium.com/catalog/product/1092167>

3. Немцова Т.И., Голова С.Ю. Программирование на языке высокого уровня. Программирование на языке С++ [Электронный ресурс]: Учебное пособие. - Москва: Издательский Дом "ФОРУМ", 2021. - 512 – Режим доступа: <https://znanium.com/catalog/product/1172261>

4. Кислицын Е. В., Шишков Е. И. Разработка приложений на языке Java [Электронный ресурс]: учебное пособие. - Екатеринбург: [Издательство УрГЭУ], 2017. - 86 – Режим доступа: <http://lib.usue.ru/resource/limit/ump/17/p488938.pdf>

5. Плещев В. В., Шишков Е. И. Основы программирования на языках С++ и С# с примерами и упражнениями [Электронный ресурс]: учебное пособие. - Екатеринбург: [Издательство УрГЭУ], 2018. - 286 – Режим доступа: <http://lib.usue.ru/resource/limit/ump/18/p490708.pdf>

10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ОНЛАЙН КУРСОВ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Перечень лицензионного программного обеспечения:

Astra Linux Common Edition. Договор №0417-ПО/2019 от 08.05.2019, Акт №Sk000343 от 24.05.2019 и Контракт № 35-У/2018 от 13.06.2018, Акт № УТ213 от 17.12.2018. Срок действия лицензии - без ограничения срока.

Microsoft Visual Studio Community. Лицензия для образовательных учреждений. Срок действия лицензии - без ограничения срока.

Язык программирования Python. Python Software Foundation License (PSFL). Срок действия лицензии - без ограничения срока.

Microsoft Windows 10 .Договор № 52/223-ПО/2020 от 13.04.2020, Акт № Tr000523459 от 14.10.2020. Срок действия лицензии - Без ограничения срока.

Microsoft Office 2016. Договор № 52/223-ПО/2020 от 13.04.2020, Акт № Tr000523459 от 14.10.2020 Срок действия лицензии - Без ограничения срока.

МойОфис стандартный. Соглашение № СК-281 от 7 июня 2017. Дата заключения - 07.06.2017. Срок действия лицензии - без ограничения срока.

Перечень информационных справочных систем, ресурсов информационно-телекоммуникационной сети «Интернет»:

Справочно-правовая система Гарант. Договор № 58419 от 22 декабря 2015. Срок действия лицензии -без ограничения срока

Справочно-правовая система Консультант +. Договор № 143/223-У/2025 от 02.12.2025 Срок действия лицензии до 31.12.2026

Программирование и разработка веб-приложений

<https://openedu.ru/course/ITMOUniversity/PWADEV/>

Программирование на C#

<https://openedu.ru/course/urfu/CSHARP/>

11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Реализация учебной дисциплины осуществляется с использованием материально-технической базы УрГЭУ, обеспечивающей проведение всех видов учебных занятий и научно-исследовательской и самостоятельной работы обучающихся:

Специальные помещения представляют собой учебные аудитории для проведения всех видов занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду УрГЭУ.

Все помещения укомплектованы специализированной мебелью и оснащены мультимедийным оборудованием спецоборудованием (информационно-телекоммуникационным, иным компьютерным), доступом к информационно-поисковым, справочно-правовым системам, электронным библиотечным системам, базам данных действующего законодательства, иным информационным ресурсам служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа презентации и другие учебно-наглядные пособия, обеспечивающие тематические иллюстрации.

7.3.1. Примерные вопросы для самостоятельной подготовки к зачету/экзамену

К экзамену

1. Понятие класса. Объявление класса.
2. Статические и динамические поля.
3. Методы. Статические и динамические методы.
4. Методы расширения.
5. Статические классы.
6. Наследование: понятие, иерархия.
7. Класс Array.
8. Понятие интерфейса.
9. Интерфейс IComparable. Реализация интерфейса.
10. Интерфейс IComparer. Реализация интерфейса.
11. Полиморфизм. Перегрузка и переопределение методов. Виртуальные методы. Ключевые слова virtual и override.
12. Ключевое слово private. Свойства (get, set).
13. Конструкторы.
14. Поля readonly и статические конструкторы.
15. Структуры. Инициализация полей структуры.
16. Ключевые слова ref и out.
17. Структуры данных: стек, очередь, дек.
18. Реализация очереди на связных списках.
19. Понятие «апкаст» и «даункаст».
20. Дженерик-классы.
21. Использование стека для анализа скобочных выражений.
22. Использование очереди для нахождения скользящего среднего.
23. Использование дженерик-методов для создания универсальной сортировки.
24. Дженерик-классы Tuple и Nullable.
25. Интерфейсы IEnumerable и IEnumerator. Реализация интерфейсов.
26. Реализация метода GetEnumerator с помощью конструкции yield return.
27. Списки. Индексация в списках.
28. Реализация методов Contains и Equals.
29. Перегрузка операторов.
30. Хеш-функции. Виртуальный метод GetHashCode.
31. Алгоритм нахождения подстроки в строке с помощью хеш-функции.
32. Делегаты: понятие и свойства.
33. Дженерик-делегаты.
34. Конструкции Func и Action. Анонимные делегаты.
35. Лямбда-выражения.
36. Язык LINQ. Реализация метода Where.
37. Язык LINQ. Реализация методов Select и ToList.
38. Язык LINQ. Методы фильтрации и преобразования.
39. Язык LINQ. Функции агрегирования и группировки.
40. События. Определение и вызов событий. Обработчики событий.
41. Использование делегатов для обработки событий.
42. Мультикаст-делегаты.
43. Назначение protected virtual OnTick, sender и EventArgs.
44. WindowsForm и WPF
45. Элементы управления.
46. Реализация резинового дизайна. Менеджеры компоновки.

47. Рисование.
48. Таймеры и анимация.
49. Паттерн MVC.
50. Потoki, домены и процессы. Класс Thread.
51. Методы BeginInvoke и EndInvoke для управления потоками.
52. Класс Parallel для асинхронного выполнения функций. Общие ресурсы и lock.
53. Блокирование потока с графическим интерфейсом. Асинхронные операции в приложениях с GUI.
54. Конструкции async и await. Класс BackgroundWorker.
55. Рефлексия типов.

7.3.2. Практические задания по дисциплине для самостоятельной подготовки к экзамену

Примерные практические задания к экзамену (ОПК-3, ОПК-4)

1. Разработать Windows-приложение, строящее диаграмму или график по заданным в файле данным. Создать меню с командами Input Data, Choose, Line, Bar, Quit. Команды Line и Bar недоступны. При выборе команды Input Data из файла читаются исходные данные. По команде Choose открывается диалоговое окно, содержащее: список для выбора цвета графика типа TListBox; группу из двух переключателей (Line, Bar) типа RadioButton; кнопку типа Button. Обеспечить возможность ввода цвета и выбора режима: график или диаграмма. После указания параметров становится доступной соответствующая команда меню. По команде Line или Bar в главном окне приложения выбранным цветом строится график или диаграмма. Окно должно содержать заголовок графика или диаграммы, наименование и градацию осей. Изображение должно занимать все окно и масштабироваться при изменении размеров окна.
2. Описать класс для работы с одномерным массивом целых чисел. Обеспечить следующие возможности: 1. задание произвольных целых границ; 2. обращение к отдельному элементу массива с контролем выхода за пределы массива; 3. Выполнение операций поэлементного сложения и вычитания массивов с одинаковыми границами индексов; 4. Выполнение операций умножения и деления всех элементов массива на скаляр; 5. Вывод на экран элемента массива по заданному индексу и всего массива. Написать программу, демонстрирующую все разработанные элементы класса.
3. Описать класс, представляющий треугольник. Предусмотреть методы для создания объектов, перемещения на плоскости, изменения размеров и вращения на заданный угол. Описать свойства для получения состояния объекта. При невозможности построения треугольника выбрасывается исключение. Написать программу, демонстрирующую все разработанные элементы класса.
4. Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел. Написать программу, демонстрирующую все разработанные элементы класса.
5. Составить описание класса для вектора, заданного координатами его концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами. Написать программу, демонстрирующую все разработанные элементы класса.
6. Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменение размеров, построение наименьшего прямоугольника, содержащего два заданных прямоугольника, и прямоугольника, являющегося общей частью (пересечением) двух прямоугольников. Написать программу, демонстрирующую все разработанные элементы класса.
7. Составить описание класса для представления времени. Предусмотреть возможности установки времени и изменения его отдельных полей (час, минута, секунда) с проверкой допустимости вводимых значений. В случае недопустимых значений полей выбрасываются

- ся исключения. Создать метод `ы` изменения времени на заданное количество часов, минут и секунд. Написать программу, демонстрирующую все разработанные элементы класса.
8. Описать класс, представляющий треугольник. Предусмотреть методы для создания объектов, вычисления площади, периметра и точки пересечения медиан. Описать свойства для получения состояния объекта. При невозможности построения треугольника выбрасывается исключение. Написать программу, демонстрирующую все разработанные элементы класса.
 9. Описать класс, представляющий круг. Предусмотреть методы для создания объектов, вычисления площади круга, длины окружности и проверки попадания заданной точки внутрь круга. Описать свойства для получения состояния объекта. Написать программу, демонстрирующую все разработанные элементы класса.
 10. Описать класс дробей — рациональных чисел, являющихся отношением двух целых чисел. Предусмотреть методы сложения, вычитания, умножения и деления дробей. Написать программу, демонстрирующую все разработанные элементы класса.
 11. Описать класс «комната», содержащий сведения о метраже, высоте потолков количестве окон. Предусмотреть инициализацию с проверкой допустимости значений полей. В случае недопустимых значений полей выбрасываются исключения. Описать методы вычисления площади и объема комнаты и свойства для получения состояния объекта. Написать программу, демонстрирующую все разработанные элементы класса.
 12. Описать класс, представляющий квадратное уравнение вида $ax^2 + vx + c = 0$. Описать метод, вычисляющий решение этого уравнения и выбрасывающий исключение в случае отсутствия корней. Описать свойства для получения состояния объекта. Написать программу, демонстрирующую все разработанные элементы класса.
 13. Описать класс «процессор», содержащий сведения о марке, тактовой частоте, объеме кэша и стоимости. Предусмотреть инициализацию с проверкой допустимости значений полей. В случае недопустимых значений полей выбрасываются исключения. Описать свойства для получения состояния объекта. Описать класс «материнская плата», включающий класс «процессор» и объем установленной оперативной памяти. Предусмотреть инициализацию с проверкой допустимости значений поля объема памяти. В случае недопустимых значений поля выбрасывается исключение. Описать свойства для получения состояния объекта. Написать программу, демонстрирующую все разработанные элементы классов.
 14. Описать класс «записная книжка». Предусмотреть возможность работы с произвольным числом записей, поиска записи по какому-либо признаку (например, по фамилии, дате рождения или номеру телефона), добавления и удаления записей, сортировки по фамилии и доступа к записи по номеру. Написать программу, демонстрирующую все разработанные элементы класса.
 15. Описать класс «студенческая группа». Предусмотреть возможность работы с переменным числом студентов, поиска студента по какому-либо признаку (например, по фамилии, имени, дате рождения), добавления и удаления записей, сортировки по разным полям, доступа к записи по номеру. Написать программу, демонстрирующую все разработанные элементы класса.
 16. Описать класс «колода карт», включающий закрытый массив элементов класса «карта». В карте хранятся масть и номер. Обеспечить возможность вывода карты по номеру, вывода всех карт, перемешивания колоды и выдачи всех карт из колоды поодиночке и по 6 штук в

случайном порядке. Написать программу, демонстрирующую все разработанные элементы классов.

17. Описать класс «поезд», содержащий следующие закрытые поля: название пункта назначения; номер поезда (может содержать буквы и цифры); время отправления. Предусмотреть свойства для получения состояния объекта. Описать класс «вокзал», содержащий закрытый массив поездов. Обеспечить следующие возможности: вывод информации о поезде по номеру с помощью индекса; вывод информации о поездах, отправляющихся после введенного с клавиатуры времени; перегруженную операцию сравнения, выполняющую сравнение времени отправления двух поездов; вывод информации о поездах, отправляющихся в заданный пункт назначения. Информация должна быть отсортирована по времени отправления. Написать программу, демонстрирующую все разработанные элементы классов.
18. Описать класс «товар», содержащий следующие закрытые поля: название товара; название магазина, в котором продается товар; стоимость товара в рублях. Предусмотреть свойства для получения состояния объекта. Описать класс «склад», содержащий закрытый массив товаров. Обеспечить следующие возможности: вывод информации о товаре по номеру с помощью индекса; вывод на экран информации о товаре, название которого введено с клавиатуры; если таких товаров нет, выдать соответствующее сообщение; сортировку товаров по названию магазина, по наименованию и по цене; перегруженную операцию сложения товаров, выполняющую сложение их цен. Написать программу, демонстрирующую все разработанные элементы классов.
19. Описать класс «самолет», содержащий следующие закрытые поля: название пункта назначения; шестизначный номер рейса; время отправления. Предусмотреть свойства для получения состояния объекта. Описать класс «аэропорт», содержащий закрытый массив самолетов. Обеспечить следующие возможности: вывод информации о самолете по номеру рейса с помощью индекса; вывод информации о самолетах, отправляющихся в течение часа после введенного с клавиатуры времени; вывод информации о самолетах, отправляющихся в заданный пункт назначения; перегруженную операцию сравнения, выполняющую сравнение времени отправления двух самолетов. Информация должна быть отсортирована по времени отправления. Написать программу, демонстрирующую все разработанные элементы классов.
20. Описать класс «запись», содержащий следующие закрытые поля: фамилия, имя; номер телефона; дата рождения (массив из трех чисел). Предусмотреть свойства для получения состояния объекта. Описать класс «записная книжка», содержащий закрытый массив записей. Обеспечить следующие возможности: вывод на экран информации о человеке, номер телефона которого введен с клавиатуры; если такого нет, выдать на дисплей соответствующее сообщение; поиск людей, день рождения которых сегодня или в заданный день; поиск людей, день рождения которых будет на следующей неделе; поиск людей, номер телефона которых начинается на три заданных цифры. Написать программу, демонстрирующую все разработанные элементы классов.
21. Описать базовый класс «Строка». Обязательные поля класса: поле для хранения символов строки; значение типа `word` для хранения длины строки в байтах. Реализовать обязательные методы следующего назначения: конструктор без параметров; конструктор, принимающий в качестве параметра строковый литерал; конструктор, принимающий в качестве параметра символ; метод получения длины строки; метод очистки строки. Описать производный от «Строка» класс «Комплексное число». Строки данного класса состоят из двух

полей, разделенных символом i . Первое поле задает значение действительной части числа, второе – значение мнимой. Каждое из полей может содержать только символы десятичных цифр и символы $+$ и $-$, задающие знак числа. Символы $+$ и $-$ могут находиться только в первой позиции числа, причем символ $+$ может отсутствовать, в этом случае число считается положительным. Для класса «Комплексное_число» определить следующие методы: проверка на равенство; сложение чисел; умножение чисел.

22. Создать меню с командами Begin, Work, About. При выборе команды About открывается окно с информацией о разработчике. При выборе команды Begin открывается диалоговое окно, содержащее: два поля ввода; группу из двух переключателей (First letter, All letter) типа RadioButton; кнопку типа Button. Обеспечить возможность ввода предложения и выбора режима его преобразования: либо начинать с прописной буквы каждое слово (First letter), либо перевести все буквы в верхний регистр (All letter). При выборе команды Work открывается диалоговое окно с результатами преобразования.
23. Написать Windows-приложение – графическую иллюстрацию сортировки методом выбора. Создать меню с командами File, Animate, About, Exit. Команда Animate недоступна. Команда Exit завершает работу приложения. Команда About открывает окно с информацией о разработчике. Для выбора файла исходных данных (команда File) использовать объект класса OpenFileDialog. Из выбранного файла читаются исходные данные для сортировки (сформировать самостоятельно не менее трех файлов различной длины с данными целого типа). После чтения данных становится доступной команда Animate. При выборе команды Animate в главном окне приложения отображается процесс сортировки в виде столбиковой диаграммы. Каждый элемент представляется столбиком соответствующего размера. На каждом шаге алгоритма два элемента меняются местами. Окно должно содержать заголовок. Изображение должно занимать все окно.