МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Волгоградский государственный технический университет»

Факультет электроники и вычислительной техники

УТВЕРЖДЕНО
Факультет электроники и вычислительной техники
Декан Авдеюк О.А.
г.

Введение в разработку программного обеспечения

рабочая программа дисциплины (модуля, практики)

Закреплена за кафедрой **Электронно-вычислительные машины и системы**

Учебный план Направление 09.03.01 Информатика и вычислительная техника
Профиль "Вычислительные машины, комплексы, системы и сети"

Профиль

Квалификация **Бакалавр**

Срок обучения **4 года**

Форма обучения **очная**

Общая трудоемкость **3 ЗЕТ**

Виды контроля в семестрах: экзамены 7

Семестр(Курс.Номер семестра на курсе)	**7(4.1)**		Итого	
	УП	ПП	УП	ПП
Лекции	16	16	16	16
Лабораторные	32	32	32	32
Итого ауд.	48	48	48	48
Контактная работа	48.35	48.35	48.35	48.35
Сам. работа	24	24	24	24
Часы на контроль	35.65	35.65	35.65	35.65
Практическая подготовка	0	0	0	0
Итого трудоемкость в часах	108	108	0	0

ЛИСТ ОДОБРЕНИЯ, СОГЛАСОВАНИЯ И АКТУАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ

Разработчик(и) программы:

зав. каф. Андреев Андрей Евгеньевич ктн

Рецензент(ы):
(при наличии)

Рабочая программа дисциплины (модуля, практики)
Введение в разработку программного обеспечения

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 929)

составлена на основании учебного плана:

Направление 09.03.01 Информатика и вычислительная техника
Профиль "Вычислительные машины, комплексы, системы и сети"

Профиль:

утвержденного учёным советом вуза от 05.06.2019 протокол № 12.

Рабочая программа одобрена на заседании кафедры
Электронно-вычислительные машины и системы

номер протокола 2019 г.
Зав. кафедрой Андреев Андрей Евгеньевич

СОГЛАСОВАНО:
Факультет электроники и вычислительной техники
Председатель НМС факультета: Авдеюк О.А.

Протокол заседания НМС от
г. №

Рабочая программа дисциплины (модуля, практики) актуализирована 31.08.2023

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ). ВИД, ТИП ПРАКТИКИ, СПОСОБ И ФОРМА (ФОРМЫ) ЕЕ ПРОВЕДЕНИЯ.
Целью дисциплины является освоение основных технологических процессов проектирования и разработки программного обеспечения (ПО) в рамках объектно-ориентированного итеративного адаптивного подхода.
Основными задачами дисциплины являются :
- изучение процессов проектирования и разработки ПО;
- использование декомпозиции и абстракции при проектировании;
- освоение итеративного планирования и разработки;
- освоение методов тестирования и отладки программного обеспечения;
- освоение процесса переработки программного обеспечения;
- изучение шаблонов проектирования и методов упаковки приложений.

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ	
Цикл (раздел) ОП:	Б1.В
2.1	**Требования к предварительной подготовке обучающегося:**
2.1.1	Мобильные и встраиваемые операционные системы
2.1.2	Основы системного программного обеспечения
2.1.3	Базы данных
2.1.4	Основы программирования
2.2	**Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:**
2.2.1	Производственная практика: Преддипломная практика
2.2.2	Выполнение и защита выпускной квалификационной работы

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)
ОПК-8: Способен разрабатывать алгоритмы и программы, пригодные для практического применения;
ОПК-8.1: Знать: методологию и основные методы математического моделирования, классификацию и условия применения моделей, основные методы и средства проектирования информационных и автоматизированных систем, инструментальные средства моделирования и проектирования информационных и автоматизированных систем.
Результаты обучения: Результаты обучения : Знает основы унифицированного языка моделирования UML, принципы объектного моделирования и проектирования, анализа вариантов использования, шаблоны работы с базами данных и разработки интерфейса пользователя
ОПК-8.2: Уметь: применять на практике математические модели, методы и средства проектирования и автоматизации систем на практике.
Результаты обучения: Результаты обучения : Умеет создавать модели программных систем с использованием UML, описывать варианты использования систем, проектировать и реализовывать компоненты программного обеспечения с использованием шаблонов проектирования
ОПК-8.3: Иметь навыки: моделирования и проектирования информационных и автоматизированных систем.
Результаты обучения: Результаты обучения : имеет навыки моделирования информационных и автоматизированных систем средствами UML и проектирования информационных и автоматизированных систем с использованием объектного подхода и шаблонов проектирования.
ПК-1: Способен оценивать качество программного обеспечения, разрабатывать стратегии тестирования, тестовые примеры и сценарии для ручного и автоматизированного тестирования посредством специализированных инструментов, анализировать полученные результаты и составлять отчетную документацию
ПК-1.1: Знать: формальные методы, технологии и инструменты для оценивания программного обеспечения; концепции и стратегии тестирования, тестовые примеры и сценарии для ручного и автоматизированного тестирования:
Результаты обучения: Результаты обучения : Знает методы, технологии и инструменты для оценивания и модульного тестирования программного обеспечения, шаблоны и приемы тестирования.
ПК-1.2: Уметь: оценивать программное обеспечение, разрабатывать стратегии тестирования, тестовые примеры и сценарии для ручного и автоматизированного тестирования посредством специализированных инструментов, анализировать полученные результаты и составлять отчетную документацию.
Результаты обучения: Результаты обучения : умеет оценивать программное обеспечение, разрабатывать стратегию и выполнять модульное тестирование программного обеспечения, применять шаблоны и приемы тестирования.
ПК-1.3: Владеть навыком: оценки качества программного обеспечения, разработок стратегий тестирования, тестовых примеров и сценарий для ручного и автоматизированного тестирования посредством специализированных инструментов, анализа полученных результатов и составления отчетной документации.
Результаты обучения: Результаты обучения : владеет навыками оценки программного обеспечения, расчета метрик ПО, разработки стратегии и покрытия тестами программного обеспечения, применения библиотек тестирования.

ПК-3: Способен проектировать программное обеспечение и базы данных, разрабатывать и отлаживать программный код, преобразовать программный код и структуры данных для повышения их эффективности, проводить рефакторинг программных приложений и баз данных
ПК-3.1: Знать: методы проектирования программного обеспечения и баз данных, разработки и настройки программного кода, преобразований программного кода и структуры данных для повышения их эффективности, проведения рефакторинга программных приложений и баз данных.
Результаты обучения: Результаты обучения : знает принципы объектного моделирования и проектирования, шаблоны объектного проектирования, шаблоны работы с базами данных и разработки интерфейса пользователя, принципы и приемы рефакторинга программных приложений и структур данных с использованием модульных тестов.
ПК-3.2: Уметь: проектировать программное обеспечение и базы данных, разрабатывать и отлаживать программный код, преобразовать программный код и структуры данных для повышения их эффективности, проводить рефакторинг программных приложений и баз данных.
Результаты обучения: Результаты обучения : умеет применять принципы объектного моделирования и проектирования, шаблоны объектного проектирования, шаблоны работы с базами данных и разработки интерфейса пользователя, проводить разработку и рефакторинг программных приложений и структур данных с использованием модульных тестов.
ПК-3.3: Владеть навыком: проектирования программного обеспечения и базы данных, разработок и настройки программного кода, преобразовывать программный код и структуры данных для повышения их эффективности, провождения рефакторинга программных приложений и баз данных.
Результаты обучения: Результаты обучения : владеет навыками проектирования программного обеспечения и баз данных, разработки и настройки программного кода, проведения рефакторинга программных приложений и баз данных; владеет современными инструментальными средствами разработки, такими как интегрированные среды, оболочки модульного тестирования, браузеры рефакторинга, средства UML моделирования; владеет навыками упаковки программного обеспечения
ПК-7: Разработка требований и проектирование программного обеспечения
ПК-7.1: Знать: методы и способы анализа и разработки требований, методы проектирования программного обеспечения
Результаты обучения: Результаты обучения : знает основы унифицированного языка моделирования UML, принципы объектного моделирования и проектирования и анализа вариантов использования
ПК-7.2: Уметь: анализировать и разрабатывать требования к программногоу обеспечению, проектировать программное обеспечение
Результаты обучения: Результаты обучения : умеет создавать модели программных систем с использованием UML, описывать варианты использования программных систем, проектировать программные системы с использованием шаблонов.
ПК-7.3: Владеть: навыками анализа и разработки требований и проектирования программного обеспечения
Результаты обучения: Результаты обучения : умеет анализировать и разрабатывать требования и проектировать ПО в рамках итеративного подхода

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Форма контроля
1	**Раздел 1. Обучение**			
1.1	ТЕХНОЛОГИИ И ТЕХНОЛОГИЧЕСКИЕ ПОДХОДЫ К РАЗРАБОТКЕ ПРОГРАММ. /Тема/	7	0	
1.1.1	Понятие о технологиях программирования, технологических подходах, жизненном цикле разработки программного обеспечения (ПО). Основные модели жизненного цикла разработки ПО и технологические подходы к разработке. Гибкие технологические подходы. Основные технологические процессы в гибких подходах. Средства коллективной разработки и контроля версий. /Лек/	7	2	К,Ко,Эк
1.1.2	Разработка адаптируемых многослойных приложений с применением технологий гибкой разработки (рефакторинг, модульные тесты, паттерны проектирования). Системы контроля версий. /Лаб/	7	4	К,Ко,Эк
1.2	ОСНОВЫ ОБЪЕКТНОГО ПОДХОДА К ПРОЕКТИРОВАНИЮ И РАЗРАБОТКЕ ПО. УНИФИЦИРОВАННЫЙ ЯЗЫК МОДЕЛИРОВАНИЯ UML. /Тема/	7	0	
1.2.1	Понятие объекта и класса, инкапсуляция, наследование, полиморфизм. Классификация предметной области. Принципы объектного проектирования. Виды диаграмм UML и их использование /Лек/	7	2	К,Ко,Эк
1.3	АНАЛИЗ ТРЕБОВАНИЙ И ПЛАНИРОВАНИЕ В ГИБКИХ ТЕХНОЛОГИЧЕСКИХ ПОДХОДАХ. /Тема/	7	0	
1.3.1	Понятие о прецедентном анализе. UML как инструмент представления требований. Итеративное планирование. /Лек/	7	2	К,Ко,Эк
1.3.2	Сеанс анализа требований и итеративного планирования. /Лаб/	7	4	К,Ко,Эк
1.4	СРЕДСТВА РАЗРАБОТКИ, CASE СРЕДСТВА И MDD. /Тема/	7	0	

1.4.1	Инструментальные средства и среды разработки. Понятие о разработке, управляемой моделью (MDD). CASE и RAD средства. /Лек/	7	2	К,Ко,Эк
1.4.2	Знакомство с разработкой, управляемой моделью (MDD), с применением UML. /Лаб/	7	4	К,Ко,Эк
1.5	ТЕСТИРОВАНИЕ И ПЕРЕРАБОТКА ПО. /Тема/	7	0	
1.5.1	Виды тестов. Модульное тестирование. Оболочки модульного тестирования. Переработка кода. Шаблоны тестирования и переработки. Понятие о разработке, управляемой тестами (TDD). /Лек/	7	2	К,Ко,Эк
1.5.2	Знакомство с разработкой, управляемой тестами (TDD). Модульные тесты и рефакторинг. Контроль версий. /Лаб/	7	8	К,Ко,Эк
1.6	ТИПОВЫЕ РЕШЕНИЯ (ШАБЛОНЫ), ИСПОЛЬЗУЕМЫЕ ПРИ РАЗРАБОТКЕ ПО. /Тема/	7	0	
1.6.1	Виды и уровни шаблонов. Шаблоны проектирования классов GoF. Другие типы шаблонов. /Лек/	7	4	К,Ко,Эк
1.6.2	Паттерны проектирования классов (GoF). Рефакторинг и паттерны. /Лаб/	7	4	К,Ко,Эк
1.6.3	Паттерны корпоративных приложений на примере каркасов разработки ПО (ORM, Репозиторий, MVC и другие) /Лаб/	7	4	К,Ко,Эк
1.7	УПАКОВКА ПРИЛОЖЕНИЙ. /Тема/	7	0	
1.7.1	Принципы упаковки приложений. Метрики упаковки. /Лек/	7	2	К,Ко,Эк
1.7.2	Упаковка приложения. Паттерны и упаковка. /Лаб/	7	4	К,Ко,Эк
2	**Раздел 2. Самостоятельная работа студентов**			
2.1	в том числе /Тема/	7	0	
2.1.1	Подготовка к отчету лабораторных работ /Ср/	7	10	Ко
2.1.2	Выполнение контрольной работы /Ср/	7	14	К
3	**Раздел 3. Промежуточная аттестация**			
3.1	в том числе /Тема/	7	0	
3.1.1	/Экзамен/ /Экзамен/	7	35.65	Эк
3.1.2	Контактная работа с ППС /КоРа/	7	0.35	Эк

Примечание. Формы контроля: Эк – экзамен, К- контрольная работа, Ко- контрольный опрос, Сз- семестровое задание, З-зачет, ОП-отчет по практике.

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Оценочные средства планируемых результатов обучения представлены в виде фондов оценочных средств (ФОС), разработанных в соответствии с локальным нормативным актом университета. ФОС может быть представлен в Приложении к рабочей программе.

5.1 Контрольные вопросы и задания
Оценочные средства планируемых результатов обучения представлены в виде фондов оценочных средств (ФОС), разработанных в соответствии с локальным нормативным актом университета. В целях освоения компетенций, указанных в рабочей программе дисциплины, предусмотрены следующие вопросы, задания текущего контроля:

ОПК-8: Способен разрабатывать алгоритмы и программы, пригодные для практического применения

ОПК-8.1: Знать: методологию и основные методы математического моделирования, классификацию и условия применения моделей, основные методы и средства проектирования информационных и автоматизированных систем, инструментальные средства моделирования и проектирования информационных и автоматизированных систем.

Студент знает основы унифицированного языка моделирования UML, принципы объектного моделирования и проектирования, анализа вариантов использования, шаблоны работы с базами данных и разработки интерфейса пользователя

Вопросы, задания:

1 Перечислите принципы SOLID
2 Опишите процесс классификации (выделения классов) по описанию постановки задачи и вариантам использования
3 Назовите основные типы диаграмм UML
4 Сформулируйте и поясните принцип одиночной ответственности класса
5 Сформулируйте и поясните принцип открытия - закрытия
6 Перечислите шаблоны (паттерны) работы с базами данных
7 Перечислите шаблоны организации интерфейса пользователя
8 Какие паттерны GoF используются при работе с БД ?
9 Какие паттерны GoF используются при работе с UI ?

ОПК-8.2: Уметь: применять на практике математические модели, методы и средства проектирования и автоматизации систем.

Студент умеет создавать модели программных систем с использованием UML, описывать варианты использования систем, проектировать и реализовывать компоненты программного обеспечения с использованием шаблонов проектирования

Вопросы, задания:
1. Приведите примеры сжатого, развернутого и детального описания прецедента.
2. Приведите примеры использования разных типов UML диаграмм при описании процедентов.
3. Приведите пример итеративного плана разработки какого-либо сценария прецедента.
4. Приведите пример диаграммы последовательности.
5. Разработайте реализацию прецедента с использованием паттернов проектирования
6. Приведите пример применения паттерна Фасад
7. Приведите пример применения паттерна Наблюдатель
8. Приведите пример применения паттерна Стратегия

ОПК-8.3: Иметь навыки: моделирования и проектирования информационных и автоматизированных систем.

Студент владеет навыками моделирования информационных и автоматизированных систем средствами UML и проектирования информационных и автоматизированных систем с использованием объектного подхода и шаблонов проектирования.

Вопросы, задания:
1. Опишите детально сжатый прецедент и выделите по описанию классы.
Составьте диаграмму классов UML.
2. Разработайте интерфейсы классы для реализации прецедента с использованием диаграммы последовательности.
3. Выберите и примените паттерн проектирования для решения поставленной задачи.
4 Разработайте программную реализацию с использованием заданного паттерна проектирования для заданной задачи.

ПК-1: Способен оценивать качество программного обеспечения, разрабатывать стратегии тестирования, тестовые примеры и сценарии для ручного и автоматизированного тестирования посредством специализированных инструментов, анализировать полученные результаты и составлять отчетную документацию

ПК-1.1: Знать: формальные методы, технологии и инструменты для оценивания программного обеспечения; концепции и стратегии тестирования, тестовые примеры и сценарии для ручного и автоматизированного тестирования

Студент знает методы, технологии и инструменты для оценивания и модульного тестирования программного обеспечения, шаблоны и приемы тестирования.

Вопросы, задания:

1 Для чего предназначены модульные тесты ?
2 Как выполняется изоляция модуля при тестировании ?
3 Перечислите приемы модульного тестирования для быстрого срабатывания тестов.
4 Назовите разделы каталога шаблонов тестирования
5 Опишите принцип TDD

ПК-1.2: Уметь: оценивать программное обеспечение, разрабатывать стратегии тестирования, тестовые примеры и сценарии для ручного и автоматизированного тестирования посредством специализированных инструментов, анализировать полученные результаты и составлять отчетную документацию.

Студент умеет оценивать программное обеспечение, разрабатывать стратегию и выполнять модульное тестирование программного обеспечения, применять шаблоны и приемы тестирования.

Вопросы, задания:

1. Опишите, как Вы будете покрывать модульными тестами подсистему в проекте, который Вам дали сопровождать, но в нем нет тестов ?
2. Приведите пример создания модульного теста с использованием заданной библиотеки модульного тестирования
3. Приведите пример реализации приема тестирования Подделка
4. Приведите пример реализации приема тестирования Подставной объект
5. Приведите пример реализации приема тестирования Триангуляция

ПК-1.3: Владеть навыком: оценки качества программного обеспечения, разработок стратегий тестирования, тестовых примеров и сценарий для ручного и автоматизированного тестирования посредством специализированных инструментов, анализа полученных результатов и составления отчетной документации.

Студент владеет навыками оценки программного обеспечения, расчета метрик ПО, разработки стратегии и покрытия тестами программного обеспечения, применения библиотек тестирования.

Вопросы, задания:

1. Создайте модульный тест для тестирования заданного метода класса.
2. Создайте модульный тест с применением библиотеки подставных объектов
3. Создайте модульный тест для тестирования заданного метода класса с использованием библиотеки модульного тестирования.
4. Напишите модульные тесты для выявления и локализации ошибки
5. Напишите тесты для сопровождения заданного рефакторинга
6. Расчитайте метрики упаковки для предоставленного проекта

ПК-3: Способен проектировать программное обеспечение и базы данных, разрабатывать и отлаживать программный код, преобразовать программный код и структуры данных для повышения их эффективности, проводить рефакторинг программных приложений и баз данных

ПК-3.1: Знать: методы проектирования программного обеспечения и баз данных, разработки и настройки программного кода, преобразований программного кода и структуры данных для повышения их эффективности, проведения рефакторинга программных приложений и баз данных.

Студент знает принципы объектного моделирования и проектирования, шаблоны объектного проектирования, шаблоны работы с базами данных и разработки интерфейса пользователя, принципы и приемы рефакторинга программных приложений и структур данных с использованием модульных тестов.

1. Что такое рефакторинг и для чего он нужен ?
2. Для чего предназначены модульные тесты ?
3. Как применяются модульные тесты в процессе рефакторинга ?
4. Перечислите известные Вам паттерны для работы с базами данных
5. Перечислите известные Вам паттерны, применяемые при разработке интерфейсов пользователя
6. Какие принципе объектного проектирования используются при работе с источниками данных ?

ПК-3.2: Уметь: проектировать программное обеспечение и базы данных, разрабатывать и отлаживать программный код, преобразовать программный код и структуры данных для повышения их эффективности, проводить рефакторинг программных приложений и баз данных.

Студент умеет применять принципы объектного моделирования и проектирования, шаблоны объектного проектирования, шаблоны работы с базами данных и разработки интерфейса пользователя, проводить разработку и рефакторинг программных приложений и структур данных с использованием модульных тестов.

Вопросы, задания:

1. Составьте список тестов для заданной модификации заданного модуля (класса, библиотеки)
2. Приведите примеры переработки кода для повышения его тестопригодности
3. Какие паттерны и рефакторинги применяются при отделение предметной области от представления.
4. Приведите пример применения принципа DIP
5. Приведите пример применения паттерна Шаблонный метод.

ПК-3.3: Владеть навыком: проектирования программного обеспечения и базы данных, разработок и настройки программного кода, преобразовывать программный код и структуры данных для повышения их эффективности, провождения рефакторинга программных приложений и баз данных.

Студент владеет навыками проектирования программного обеспечения и баз данных, разработки и настройки программного кода, проведения рефакторинга программных приложений и баз данных; владеет современными инструментальными средствами разработки, такими как интегрированные среды, оболочки модульного тестирования, браузеры рефакторинга, средства UML моделирования; владеет навыками упаковки программного обеспечения

1. Выполните адаптацию заданного кода к заданным изменениям, применив паттерны проектирования
2. Составьте модульные тесты для заданной модификации заданного модуля (класса, библиотеки)
3. Выполните переработку кода для повышения его тестопригодности
4. Отделите предметную область от представления для заданного проекта
5. Примените принцип DIP для работы с данными в проекте

6. Выполните упаковку заданного проекта.

ПК-7: Разработка требований и проектирование программного обеспечения

ПК-7.1: Знать: методы и способы анализа и разработки требований, методы проектирования программного обеспечения

Студент знает основы унифицированного языка моделирования UML, принципы объектного моделирования и проектирования и анализа вариантов использования

1. Что такое прецедентный анализ ? Какие требования он позволяет описать ?
2. Как используются прецеденты при разработке ?
3. Как уточняются требования в процессе итеративной разработки на разных фазах ?
4. Какие диаграммы UML применяются в процессе анализа прецедентов ?
5. Перечислите известные Вам принципы объектного проектирования
6. Перечислите наиболее часто используемые типы диаграмм UML и их назначение

ПК-7.2: Уметь: анализировать и разрабатывать требования к программному обеспечению, проектировать программное обеспечение

Студент умеет создавать модели программных систем с использованием UML, описывать варианты использования программных систем, проектировать программные системы с использованием шаблонов.

1. Приведите примеры сжатого, развернутого и детального описания прецедента.
2. Приведите примеры использования разных типов UML диаграмм при описании прецедентов.
3. Приведите пример итеративного плана разработки какого-либо сценария прецедента.
4. Приведите пример сочетания MDD и TDD при разработке сценария прецедента.
5. Рассмотрите этапы итеративной разработки и оптимизации кода, реализующего сценарий прецедента.

ПК-7.3: Владеть: навыками анализа и разработки требований и проектирования программного обеспечения

Студент умеет анализировать и разрабатывать требования и проектировать ПО в рамках итеративного подхода

1. Составьте несколько развернутых описаний прецедентов
2. Составьте план разработки версий и одной из итераций
3. Составьте диаграмму вариантов использования.
4. Подведите итоги выполнения итераций и скорректируйте план следующей итерации по предоставленным данным

5.2 Темы письменных работ (контрольная работа)
На контрольную работу студенту выдается индивидуальное задание (по вариантам), заключающееся в итеративной разработке по варианту простого прикладного программного приложения с использованием изучаемых в дисциплине технологий (итеративное планирование, прецедентный анализ, UML, модульные тесты, рефакторинг, паттерны проектирования).
Работа выполняется в письменной форме в течение 8 недель с момента выдачи задания. Контрольный срок сдачи – последний месяц семестра.

Примерное содержание контрольной работы
1. Титульный лист.
2. Формулировка варианта задания.
3. Основная часть, включающая:
1) описание требований к приложению (состав функций),
2) описание используемых средств разработки, технологий, библиотечных функций и классов,
3) прецедентный анализ,
4) результаты моделирования предметной области,
5) структура базы данных приложения
6) архитектура приложения и используемые архитектурные шаблоны
7) диаграммы классов программы, диаграммы взаимодействия,
8) модульные тесты
9) описание примененных паттернов
10) описание переработки (рефакторинга) кода
11) упаковка приложения,
12) экранные формы работы приложения,
12) коды программы (в приложении).
13). Список использованных источников (включая источники Интернет).

Правила оформления курсовой работы

- работа оформляется в редакторе MS Word / OpenOffice (*.doc, *.docx, *.odt);
- листы формата А4, ориентация книжная;
- поля: левое – 2 см, остальные – по 1 см;
- шрифт – Times New Roman;
- размер шрифта 14 pt;
- междустрочный интервал – 1,5;
- абзацный отступ – 1,25 см;
- нумерация страниц сквозная, номер на первой странице не ставится;
- в конце работы необходим список использованной литературы согласно ГОСТ Р 7.0.5 – 2008;
- объем работы зависит от степени раскрытия основных пунктов курсовой работы.

Примерный список вариантов контрольной работы :

1. Простой текстовый редактор (сохранение, загрузка, копирование, вставка, отмена и т.п.).
2. Программа для форматирования исходного кода по набору правил (GUI + CUI).
3. Утилита для массового переименования файлов по набору правил (GUI + CUI).
4. Программа для подсчёта метрик исходного кода (число строк, классов, методов и т.п.).
5. Простая программа расчета заработной платы.
6. Система управления банкоматом.
7. Система ведения счёта кредитной карты (бонусы, скидки, штрафы, кредит).
8. БД Клиенты (покупки (суммы, даты) и статистика по клиентам).
9. БД Группы (студенты / предметы с оценками (средний балл и пр.) «АРМ деканата».
10. БД Конфигурации ПК (детали (расчет конфигурации)) «АРМ компьютерной фирмы».
11. БД Официанты (заказы / блюда (расчет счета)) – «АРМ Кафе».
12. БД Товары (остатки на складах (статистика по складу)) – простая система складского учета
13. БД Управления кадров (Отделы / сотрудники (статистика по отделам)) - «АРМ Кадры»
14. БД кафедры (статистика по ППС) - «АРМ Кафедры»
15. Программная система интернет-биллинга.
16. Программная система биллинга услуг ЖКХ.
17. Программная система биллинга услуг телефонной связи.
18. Программная система биллинга услуг сотовой связи.
19. Программная система управления платежным терминалом.
20. Система бронирования авиабилетов.

5.3 Показатели и критерии оценивания компетенций, шкалы оценивания
В рамках изучаемой дисциплины студент может демонстрировать следующие уровни овладения компетенциями.

Повышенный уровень: обучающийся демонстрирует глубокое знание учебного материала; способен использовать сведения из различных источников для успешного исследования и поиска решения в нестандартных ситуациях; способен анализировать, проводить сравнение и обоснование выбора методов решения практико-ориентированных заданий. Оценка промежуточной аттестации (экзамен): 5 (отлично) – 90 баллов и более.

Базовый уровень: обучающийся способен понимать и интерпретировать освоенную информацию; демонстрирует осознанное владение учебным материалом и учебными умениями, навыками и способами деятельности, необходимыми для решения практико-ориентированных заданий. Оценка промежуточной аттестации (экзамен): 4 (хорошо) – 76-89 баллов.

Пороговый уровень: обучающийся обладает необходимой системой знаний и владеет некоторыми умениями; демонстрирует самостоятельность в применении знаний, умений и навыков к решению учебных заданий на репродуктивном уровне. Оценка промежуточной аттестации (экзамен): 3 (удовлетворительно) – 61-75 баллов.

Уровень ниже порогового: система знаний, необходимая для решения учебных и практико-ориентированных заданий, не сформирована; обучающийся не владеет основными умениями, навыками и способами деятельности. Оценка промежуточной аттестации (экзамен): 2 (неудовлетворительно) – ниже 61 балла.

В рамках данной дисциплины используются следующие критерии оценки знаний студентов.

Отлично
Обучающийся демонстрирует:
- систематизированные, глубокие и полные знания по всем разделам учебной дисциплины, а также по основным вопросам, выходящим за ее пределы;
- точное использование научной терминологии, грамотное, логически правильное изложение ответа на вопросы;
- безупречное владение инструментарием учебной дисциплины, умение его эффективно использовать в постановке и решении научных и профессиональных задач;
- выраженную способность самостоятельно и творчески решать сложные проблемы в нестандартной ситуации;
- полное и глубокое усвоение основной, и дополнительной литературы, по изучаемой учебной дисциплине;
- умение свободно ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им

аналитическую оценку, использовать научные достижения других дисциплин;
- творческую самостоятельную работу на учебных занятиях, активное творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Хорошо
Обучающийся демонстрирует:
- систематизированные, глубокие и полные знания по всем разделам учебной дисциплины;
- использование научной терминологии, грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы и обобщения;
- владение инструментарием учебной дисциплины (методами комплексного анализа, техникой информационных технологий), умение его использовать в постановке и решении научных и профессиональных задач;
- способность решать сложные проблемы в рамках учебной дисциплины;
- свободное владение типовыми решениями;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой по учебной дисциплине;
- умение ориентироваться в теориях, концепциях и направлениях по изучаемой учебной дисциплине и давать им аналитическую оценку;
- активную самостоятельную работу на учебных занятиях, систематическое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Удовлетворительно
Обучающийся демонстрирует:
- достаточные знания в объеме рабочей программы по учебной дисциплине;
- использование научной терминологии, грамотное, логически правильное изложение ответа на вопросы, умение делать выводы без существенных ошибок;
- владение инструментарием учебной дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно применять типовые решения в рамках изучаемой дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой по дисциплине;
- умение ориентироваться в базовых теориях, концепциях и направлениях по дисциплине;
- работу на учебных занятиях под руководством преподавателя, фрагментарное участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Неудовлетворительно
Обучающийся демонстрирует:
- фрагментарные знания в рамках изучаемой дисциплины; знания отдельных литературных источников, рекомендованных рабочей программой по учебной дисциплине;
- неумение использовать научную терминологию учебной дисциплины, наличие в ответе грубых, логических ошибок;
- пассивность на занятиях или отказ от ответа, низкий уровень культуры исполнения заданий.

5.4. Вопросы промежуточной аттестации
1. Общая классификация технологических подходов и процессов разработки. Традиционный подход к разработке программ (каскадный или водопадный процесс) и итеративные подходы.
2. Унифицированный процесс (UP) и Rational Unified Process (RUP).
3. Принципы быстрой разработки (agile development). Манифест альянса быстрой разработки.
4. Основные принципы и практики экстремального программирования (XP). Отличия XP от каскадного и унифицированного подходов.
5. Виды требований. Функциональные и нефункциональные требования, ограничения.
6. Понятие о прецедентах. Виды описания прецедентов. Анализ прецедентов.
7. Пользовательские истории в XP. Их отличия от прецедентов.
8. Анализ дополнительных требований.
9. Планирование разработки. Экстремальное планирование. Общий план. Планы версий и итераций, текущее планирование. Измерение скорости работы.
11. Архитектурное проектирование. Задачи. Архитектурные паттерны.
12. Понятие о классе и объекте. Атрибуты (поля) и методы классов. Инкапсуляция, наследование, полиморфизм.
13. Отношения между классами и объектами. Виды отношений.
14. Ассоциация, агрегация и композиция. Отношение зависимости.
15. Наследование и реализация. Сравнение наследования и делегирования.
16. Абстрактные классы и интерфейсы. Назначение интерфейсов.
17. Задача классификации на примере анализа предметной области.
18. Базовые принципы объектного проектирования (SRP, OCP, DIP и др.)
19. Понятие об UML. Назначение UML. Виды диаграмм UML.
20. Диаграммы классов и пакетов. Диаграммы прецедентов.
21. Диаграммы взаимодействия.
22. Диаграммы активности и состояний.
23. Диаграммы компонентов и размещения.
24. Разработка, управляемая моделью (MDD). Основные принципы. Использование диаграмм UML. Пример MDD.
25. Использование CASE — средств в процессе MDD. Reverse Engineering.
26. Практики экстремального программирования, их назначение и особенности применения.

27. Понятие о модульном тестировании. Назначение, способы реализации.
28. Каталог шаблонов тестирования (виды модульных тестов).
29. Понятие о разработке через тестирование (TDD).
30. Оболочка xUnit/NUnit. Назначение, структура, применение.
31. Понятие о приемочном тестировании. Назначение, автоматизация.
32. Понятие о рефакторинге.
33. Каталог шаблонов рефакторинга.
34. Понятие о шаблонах (паттернах). Паттерны проектирования. Классификация паттернов.
35. Паттерны GoF. Классификация паттернов GoF. Назначение паттернов.
36. Шаблоны структурирования (Адаптер, Компоновщик, Фасад, Заместитель и др.)
37. Шаблоны поведения (Итератор, Наблюдатель, Стратегия, Шаблонный метод и др.)
38. Шаблоны создания (Фабричный метод, Абстрактная фабрика, Одиночка).
39. Архитектурные шаблоны (Слои, Контроллер и др.). Шаблоны создания корпоративных приложений.
40. Шаблон Слои и концепция расслоения приложений. Трех- и многоуровневые приложения.
41. Виды приложений, работающих с БД и варианты использования БД.
42. Применение шаблонов при работе с БД (Фасад, Заместитель, Активная запись, DIP, модуль таблицы, преобразователь данных-маппер, репозиторий и др.).
44. Виды пользовательского интерфейса. Роль интерфейса в приложении. Проблематика отде-ления интерфейса от основной части приложения. Анти-паттерн Smart User Interface.
45. Применение шаблонов при построении пользовательского интерфейса (Наблюдатель, Кон-троллер, Фасад и др.).
46. Задача упаковки программных проектов. Общие принципы упаковки (REP, CCP и др). Понятие о пакете.
47. Метрики упаковки. Способы улучшения метрик.
48. Понятие о распределенных приложениях. Виды распределенных приложений. Проблема-тика построения распределенных приложений.
49. Классификация технологий построения распределенных приложений.
50. Интегрированные среды (IDE) и визуальные средства быстрой разработки (RAD). Преимущества и недостатки.

5.5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Промежуточная аттестация обучающихся ведется непрерывно и включает в себя текущую аттестацию (контроль текущей работы в семестре, включая оценивание промежуточных результатов обучения по дисциплине) и семестровую аттестацию (экзамен) – оценивание окончательных результатов обучения по дисциплине.

По данной дисциплине, завершающейся экзаменом, по обязательным формам текущего контроля студенту предоставляется возможность набрать в сумме не менее 60 баллов. Оценивание окончательных результатов обучения по дисциплине ведется по 100-балльной шкале, оценка формируется автоматически как сумма количества баллов, набранных обучающимся за выполнение заданий обязательных форм текущего контроля и количества баллов, набранных на семестровой аттестации (экзамене).

Система оценивания

Текущий контроль представляет собой проверку усвоения учебного материала теоретического и практического характера, регулярно осуществляемую на протяжении семестра. К основным формам текущего контроля можно отнести устный опрос, письменные задания, лабораторные работы, контрольные работы.

Контрольная работа

Контрольная работа представляет собой законченную работу, заключающееся в итеративной разработке по варианту простого прикладного программного приложения с использованием изучаемых в дисциплине технологий (итеративное планирование, прецедентный анализ, UML, модульные тесты, рефакторинг, паттерны проектирования). Полностью выполненная контрольная работа оценивается в 20 баллов.

Лабораторная работа.

Лабораторная работа является формой контроля и средством применения и реализации полученных обучающимися знаний, умений и навыков в ходе выполнения учебно-практической задачи, связанной с получением значимого результата с помощью реальных средств деятельности. Рекомендуются для проведения в рамках тем (разделов), наиболее значимых в формировании компетенций.

Устный опрос, собеседование.

Устный опрос, собеседование являются формой оценки знаний и предполагают специальную беседу преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной. Процедуры направлены на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п. Устный ответ или собеседование может практиковаться преподавателем для уточнения знаний на лабораторных занятиях.

Устный опрос включает 1 вопрос из группы вопросов “5.1 Контрольные вопросы и задания”, собеседование может включать более 1-го вопроса того же списка. Ответ оценивается от 0 до 3 баллов следующим образом:
3 балла - полный, логически безупречный ответ;
2 балла - ответ в целом полный, но могут иметь место несущественные пробелы в знаниях; логика ответа правильная, но

некоторые моменты в своих рассуждениях студент обосновать затрудняется;
1 балл - ответ частичный, содержит значительные изъяны; нарушений логики ответа нет, но имеется ряд логических переходов в рассуждениях, которые студент обосновать затрудняется.

Промежуточная аттестация. Экзамен.
Промежуточная аттестация осуществляется в конце семестра и завершает изучение дисциплины. Промежуточная аттестация помогает оценить более крупные совокупности знаний, умений и навыков, в некоторых случаях – даже формирование определенных компетенций. В рамках данного предмета к форме промежуточного контроля относится экзамен.

Экзамен по дисциплине имеет цель оценить сформированность компетенций, теоретическую подготовку студента, его способность к творческому мышлению, приобретенные им навыки самостоятельной работы, умение синтезировать полученные знания и применять их при решении практических задач. Экзамен проводится в устной форме либо в виде тестов на компьютере. В ходе экзамена студент пишет ответ на вопросы билета. Билет включает два вопроса из списка "5.4. Вопросы промежуточной аттестации", оцениваемых по 20 баллов. При проведении тестов дается тест на 20 вопросов по тематике устного экзамена, каждый ответ оценивается в 2 балла. Дополнительные баллы, помимо баллов, полученных за курсовую работу и отчет лабораторных, могут быть заработаны за правильные ответы в ходе опросов и собеседований.

Если суммарное число баллов набранных в семестре по результатам модулей и полученных на экзамене
- от 61 до 75 , то ставится итоговая оценка "Удовлетворительно",
- от 76 до 89, то ставится итоговая оценка "Хорошо",
- от 90 до 100, то ставится итоговая оценка "Отлично".

Если суммарное число баллов, набранных студентом не менее 60 баллов, то студент может согласиться с соответствующей итоговой оценкой без экзамена.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)

6.1. Рекомендуемая литература

	Авторы, составители	Заглавие	Издательство, год.	Электронный адрес
Л.1	Камаев В. А., Костерин В. В.	Технологии программирования: учебник	Москва: Высш. шк., 2006	
Л.2	Кузнецов М. А., Андреев А. Е.	Технологии распределенных систем: современные подходы: учеб. пособие	Волгоград: ВолгГТУ, 2009	
Л.3	Андреев А. Е., Егунов В. А., Шаповалов О. В.	Технологии программирования многопроцессорных систем: учеб. пособие	Волгоград: ВолгГТУ, 2015	
Л.4	Андреев А. Е., Кирносенко С. И.	Адаптивные технологии разработки программного обеспечения: учеб. пособие	Волгоград: ВолгГТУ, 2015	
Л.5	Буч Г., Рамбо Д., Якобсон И.	Язык UML	Москва: ДМК Пресс, 2008	https://e.lanbook.com/book/1246#book_name

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	Згуральская, Е. Н. Технологии программирования : учебное пособие / Е. Н. Згуральская. — Ульяновск : УлГТУ, 2020. — 71 с. — ISBN 978-5-9795-1995-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/165011 (дата обращения: 18.09.2020). — Режим доступа: для авториз. пользователей.
Э2	Иванова, С. М. Технологии программирования. Разработка приложений на языке C# : учебное пособие / С. М. Иванова, З. В. Ильиченкова. — Москва : РТУ МИРЭА, 2021. — 73 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/176565 (дата обращения: 18.09.2021). — Режим доступа: для авториз. пользователей.
Э3	Умрихин, Е. Д. Основы разработки iOS-приложений на C# с помощью Xamarin : учебное пособие для вузов / Е. Д. Умрихин. — Санкт-Петербург : Лань, 2021. — 384 с. — ISBN 978-5-8114-6930-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/173095 (дата обращения: 18.09.2021). — Режим доступа: для авториз. пользователей.
Э4	Забродин, А. В. Основы проектирования информационных систем с помощью языка UML : учебное пособие / А. В. Забродин, В. П. Бубнов. — Санкт-Петербург : ПГУПС, 2018. — 46 с. — ISBN 978-5-7641-1133-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/111721 (дата обращения: 19.09.2020). — Режим доступа: для авториз. пользователей.
Э5	Андреев, А. Е. Адаптивные технологии разработки программного обеспечения : учебное пособие / А. Е. Андреев, С. И. Кирносенко. — Волгоград : ВолгГТУ, 2015. — 96 с. — ISBN 978-5-9948-1979-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/157223 (дата обращения: 19.09.2020). — Режим доступа: для авториз. пользователей.
Э6	Язык программирования C# и платформа .NET [Электронный ресурс] – Режим доступа : https://metanit.com/sharp/
Э7	Руководство по Django [Электронный ресурс] – Режим доступа : https://metanit.com/python/django/
Э8	Федеральный портал «Российское образование»[Электронный ресурс] – Режим доступа: www.edu.ru

Э9	Национальный Открытый Университет «ИНТУИТ»[Электронный ресурс] – Режим доступа: www.intuit.ru
6.3 Перечень программного обеспечения	
6.3.1.1	OpenOffice, LibreOffice – офисные пакеты
6.3.1.2	Microsoft Visual Studio Community – среда разработки
6.3.1.3	Microsoft Visual Studio Code – среда разработки
6.3.1.4	PyCharm Community Edition – среда разработки
6.3.1.5	Python 3 – интрепретатор и библиотеки языка программирования
6.3.1.6	Яндекс.Браузер - веб-браузер.
6.4 Перечень информационных справочных систем и электронных библиотечных систем (ЭБС)	
6.3.2.1	Библиотека (НТБ), http://library.vstu.ru/sci-nci
6.3.2.2	Электронная информационно-образовательная среда университета,http://eos2.vstu.ru
6.3.2.3	ЭБС "Лань", https://e.lanbook.com/
6.3.2.4	ЭБС "Book.ru", https://www.book.ru/
6.3.2.5	Электронная библиотека "Grebennikon", https://grebennikon.ru/
6.3.2.6	Библиографическая и реферативная база данных статей, опубликованных в научных изданиях “Scopus”, https://www.scopus.com/
6.3.2.7	Российская научная электронная библиотека, интегрированная с РИНЦ “eLIBRARY.ru”, https://www.elibrary.ru/
6.3.2.8	Поисковая интернет-платформа, объединяющая реферативные базы данных публикаций в научных журналах и
6.3.2.9	патентов “Web of Science”, https://webofknowledge.com/

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ) /ОБОРУДОВАНИЕ	
7.1	1. Мультимедийный / компьютерный класс
7.2	1) ПЭВМ Intel iCore 2ГГц / 8Гб RAM / LCD 22" - 8 шт.; 2) экран EliteScreens; 3) проектор Acer 1200;
7.3	2. Мультимедийный / компьютерный класс
7.4	1) Ноутбуки HP Elitebook 8460p – 4 шт., 2) Ноутбуки HP EliteBook 8570p - 4 шт. 3) Ноутбук Lenovo ThinkPad T420 – 4 шт. 4) экран EliteScreens;
7.5	5) проектор Acer 1203; 6) доступ в Интернет и к наукометрическим базам данных
7.6	3. Аудитория для самостоятельной работы обучающихся /Учебная мебель, компьютерная техника с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду университета (читальный зал информационно-библиотечного центра)

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ, ПРАКТИКИ)

Организация образовательного процесса по данной дисциплине регламентируется учебным планом и расписанием учебных занятий. При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет дисциплины (переаттестации ее части), если она была освоена в процессе предшествующего обучения. Перезачёт (переаттестации ее части) освобождает обучающегося от необходимости повторного освоения дисциплины (полностью или частично).
Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в электронной информационной образовательной среде.
Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана. На первой лекции лектор информирует студентов о рекомендуемой литературе и электронных источниках информации по дисциплине, с указанием, какой учебник (учебное пособие) является базовым.

Лабораторные работы предполагают выполнение и отчет заданий по темам, рассмотренным на лекционных занятиях. Каждому лабораторному занятию предшествует самостоятельная подготовка студента, включающая: ознакомление с содержанием лабораторной работы по методическим указаниям; проработку теоретической части по лекционному материалу и учебникам, рекомендованным в методических указаниях;
Самостоятельная работа студентов включает изучение законспектированного на лекционных занятиях материала, дополнение его с учетом рекомендованной по данной теме литературы, самостоятельную подготовку к лабораторным работам, самостоятельное выполнение и оформление заданий контрольной работы, аналогичных выполненным на занятиях.
В течении семестра для студентов проводятся групповые текущие консультации по учебной дисциплине и консультация перед экзаменом.

Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов
Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов

и лиц с ограниченными возможностями здоровья (ОВЗ), индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.
В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн), в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.
Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ (при необходимости).
Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств. Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания.
При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

Перечень методических указаний для освоения дисциплины представлен в разделах 6.1, 6.2, а также в учебно-методическом пособии
Андреев А.Е. Практикум по технологиям программирования : учебно-методическое пособие / Андреев А.Е., М.А.. Кузнецов,
В.Л. Абдрахманов; ВолгГТУ. – Волгоград, 2022. – 100 с.