



Федеральное государственное бюджетное образовательное учреждение высшего образования  
«ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ ДВАЖДЫ ГЕРОЯ  
СОВЕТСКОГО СОЮЗА, ЛЕТЧИКА-КОСМОНАВТА А.А. ЛЕОНОВА»

**УТВЕРЖДАЮ**

**И. о. проректора**

**А.В. Троицкий**

«\_\_» \_\_\_\_\_ 2023 г.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И  
ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**РАБОЧАЯ ПРОГРАММА  
ДИСЦИПЛИНЫ  
«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

**Направление подготовки:** 09.03.04 Программная инженерия

**Направленность (профиль):** Проектирование и разработка программного обеспечения

**Уровень высшего образования:** бакалавриат

**Форма обучения:** очная

Королев  
2023

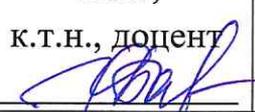
Рабочая программа является составной частью основной профессиональной образовательной программы и проходит рецензирование со стороны работодателей в составе основной профессиональной образовательной программы. Рабочая программа актуализируется и корректируется ежегодно.

**Автор: к.т.н., с.н.с. Парафейников И.В. Рабочая программа дисциплины: Объектно-ориентированное программирование – Королев МО: «Технологический университет», 2023 г.**

Рецензент: д.т.н., профессор Артюшенко В.М.

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки бакалавров 09.03.04 Программная инженерия, учебного плана, утвержденного Ученым советом Технологического университета, Протокол № 9 от 11 апреля 2023 года.

**Рабочая программа рассмотрена и одобрена на заседании кафедры:**

Заведующий кафедрой (ФИО, ученая степень, звание, подпись)	Баранова О.М., к.т.н., доцент 			
Год утверждения (переутверждения)	2023	2024	2025	2026
Номер и дата протокола заседания кафедры	№ 7 от 03.04.2023			

**Рабочая программа согласована:**

Руководитель ОПОП  к.т.н., доцент Баранова О.М.

**Рабочая программа рекомендована на заседании УМС:**

Год утверждения (переутверждения)	2023	2024	2025	2026
Номер и дата протокола заседания УМС	№ 5 от 11.04.2023 г.			

## **1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОПОП**

Целью дисциплины является формирование у обучающихся системного представления об особенностях создания программ с помощью объектно-ориентированного подхода и формирования практических навыков по созданию, отладке и модификации прикладных программ, написанных с использованием паттернов объектно-ориентированного программирования.

Дисциплина направлена на формирование следующих компетенций:

- (ОПК-2) Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;
- (ОПК-3) Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;
- (ОПК-6) Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;
- (ОПК-7) Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой.

Основными задачами дисциплины является:

- ознакомление с основными понятиями и терминологией паттернов объектно-ориентированного программирования;
- ознакомление с областями применения объектно-ориентированного программирования;
- ознакомление с концепцией объектно-ориентированного подхода применительно к бизнес-деятельности;
- выработка практических навыков в области выбора и применения паттернов объектно-ориентированного программирования.

Показатель освоения компетенции отражают следующие индикаторы:

### **Трудовые действия:**

Применяет современные информационные технологии и программные средства, в том числе, отечественного производства при решении задач профессиональной деятельности;

Применяет принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;

Анализирует информационно-коммуникационных технологий и с учетом основных требований информационной безопасности для подготовки обзоров, аннотаций, составления рефератов, научных докладов;

Применяет основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий;

Применяет в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой;

#### **Необходимые умения:**

Анализирует современные информационные технологии и программные средства, в том числе, отечественного производства при решении задач профессиональной деятельности;

Использует современные информационные технологии и программные средства, в том числе, отечественного производства, при решении задач профессиональной деятельности;

Использует знания об основах информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности для решения стандартных задач профессиональной деятельности;

Анализирует современные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды для разработки информационных систем и технологий;

Использует современные языки программирования, программные среды разработки информационных систем и технологий для отладки и тестирования прототипов программно-технических комплексов задач;

Использует основные концепции, принципы, теории и факты, связанные с информатикой для решения практических задач;

#### **Необходимые знания:**

Знает принципы работы современных информационных технологий и программных средств, в том числе отечественного производства;

Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;

Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий;

Знает основные концепции, принципы, теории и факты, связанные с информатикой.

После завершения освоения данной дисциплины студент получает:

#### **Необходимые знания:**

- основы технологии объектно-ориентированной декомпозиции программных систем, базовых шаблонов проектирования (Наблюдатель, Итератор, Одиночка, Фабрика, Заместитель), отношений

между классами и основ UML (диаграммы классов и последовательностей).

- базовые знания применения паттернов проектирования для построения программных систем.

#### **Необходимые умения:**

- применять полученные знания на практике;
- проектировать и создавать программные системы с использованием паттерном проектирования.

#### **Трудовые действия:**

- составление алгоритмов и программ на объектно-ориентированных языках программирования с применением соответствующих задачам паттернов проектирования.

## **2. Место дисциплины в структуре ОПОП ВО**

Дисциплина «Объектно-ориентированное программирование» относится к обязательной части основной профессиональной образовательной программы подготовки по направлению подготовки бакалавриата 09.03.04 «Программная инженерия».

Дисциплина реализуется кафедрой прикладного программного обеспечения.

Изучение данной дисциплины базируется на знаниях и умениях, полученных в рамках ранее изученных дисциплин «Информатика», «Основы алгоритмизации и программирования», «Информационные технологии в профессиональной деятельности», «Адаптированные информационные технологии» и компетенциях УК-2, УК-6, ОПК-2, ОПК-3, ОПК-6, ОПК-7, ОПК-8.

## **2. Объем дисциплины и виды учебной работы**

Общая трудоемкость дисциплины составляет 3 зачетные единицы, 108 часов.

**Таблица 1**

<b>Виды занятий</b>	<b>Всего часов</b>	<b>Семестр 4</b>
<b>Общая трудоемкость</b>	<b>108</b>	<b>108</b>
<b>Аудиторные занятия</b>	<b>48</b>	<b>48</b>
Лекции (Л)	<b>16</b>	<b>16</b>
Практические занятия (ПЗ)	<b>32</b>	<b>32</b>
Лабораторные работы (ЛР)	–	–
<b>Практическая подготовка</b>	-	-
<b>Самостоятельная работа</b>	<b>60</b>	<b>60</b>
Курсовые работы (проекты)	–	–
Расчетно-графические работы	–	–
Контрольная работа, домашнее задание	+ –	+ –

Текущий контроль знаний (7 - 8, 14 - 15 недели)	Тест	Тест
Вид итогового контроля	Зачёт с оценкой	Зачёт с оценкой

## 4. Содержание дисциплины

### 4.1. Темы дисциплины и виды занятий

Таблица 2

Наименование тем	Лекции, час	Практические занятия, Час	Занятия в интерактив ной форме, час	Код компетен ций
Тема 1. Введение в паттерны проектирования	2	4	2	ОПК-2, ОПК-3, ОПК-6, ОПК-7
Тема 2. Проектирование редактора документов	2	4	2	
Тема 3. Порождающие паттерны	2	4	2	
Тема 4. Структурные паттерны	2	4	2	
Тема 5. Паттерны поведения	2	4	2	
Тема 6. Реализация паттернов проектирования в языках высокого уровня	6	12	6	
<b>Итого:</b>	<b>16</b>	<b>32</b>	<b>16</b>	

### 4.2. Содержание тем дисциплины

#### Тема 1. Введение в паттерны проектирования

Введение. Основные характеристики. Описание паттернов проектирования. Каталог паттернов проектирования

#### Тема 2. Проектирование редактора документов

Задачи проектирования. Структура документа. Форматирование. Классы Compositor и Composition. Оформление пользовательского интерфейса. Паттерн декоратор. Поддержка нескольких стандартов внешнего облика. Поддержка нескольких оконных систем. Операции пользователя.

#### Тема 3. Порождающие паттерны

Паттерн Abstract Factory. Паттерн Builder. Паттерн Factory Method. Паттерн Prototype. Паттерн Singleton.

#### Тема 4. Структурные паттерны

Паттерн Adapter. Паттерн Bridge. Паттерн Composite. Паттерн Decorator. Паттерн Facade. Паттерн Flyweight. Паттерн Proxy.

## **Тема 5. Паттерны поведения**

Паттерн Chain of Responsibility. Паттерн Command. Паттерн Interpreter. Паттерн Iterator. Паттерн Mediator. Паттерн Memento. Паттерн Observer. Паттерн State. Паттерн Strategy. Паттерн Template Method. Паттерн Visitor .

## **Тема 6. Реализация паттернов проектирования в языках высокого уровня.**

C++. Python. JAVA. Диаграмма классов. Диаграмма объектов. Диаграмма взаимодействий.

### **5. Перечень учебно-методического обеспечения для самостоятельной работы по дисциплине**

Методические указания для самостоятельной работы обучающихся по освоению дисциплины представлены в Приложении 2.

### **6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине**

Структура фонда оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведена в Приложении 1.

### **7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

#### **Основная литература:**

1. Барков, И. А. Объектно-ориентированное программирование / И. А. Барков. — 2-е изд., стер. — Санкт-Петербург: Лань, 2023. — 700 с. — ISBN 978-5-507-47113-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/329549>

2. Унгер, А. Ю. Объектно-ориентированное программирование: учебное пособие / А. Ю. Унгер. — Москва: РТУ МИРЭА, 2022. — 84 с. — ISBN 978-5-7339-1628-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/265691>

3. Кувшинов, Д. Р. Основы программирования: учебное пособие для вузов / Д. Р. Кувшинов. — Москва: Издательство Юрайт, 2023. — 104 с. — (Высшее образование). — ISBN 978-5-534-07559-5. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/533784>

4. Бабушкина И. А. Практикум по объектно-ориентированному программированию: учебное пособие / И. А. Бабушкина, С. М. Окулов. — 5-е изд. — Москва: Лаборатория знаний, 2020. — 369 с. — ISBN 978-5-00101-780-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/135561>

5. Зайцев М. Г. Объектно-ориентированный анализ и программирование: учебное пособие: [16+] / М. Г. Зайцев; Новосибирский государственный технический университет. – Новосибирск: Новосибирский государственный технический университет, 2017. – 84 с.: ил., табл. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=576800>

### **Дополнительная литература:**

1. Объектно-ориентированное программирование на C++: учебник / И. В. Баранова, С. Н. Баранов, И. В. Баженова [и др.]. — Красноярск: СФУ, 2019. — 288 с. — ISBN 978-5-7638-4034-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/157572>

2. Рик Г. Объектно-Ориентированное Программирование / Г. Рик; под редакцией Н.Комлева. — Москва: СОЛОН-Пресс, 2018. — 298 с. — ISBN 978-5-91359-285-9. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/107669>

3. Чернышев С. А. Основы программирования на Python: учебное пособие для среднего профессионального образования / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2023. — 349 с. — (Профессиональное образование). — ISBN 978-5-534-17056-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/532292>

## **8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

### **Интернет-ресурсы:**

1. <https://biblioclub.ru/> - Университетская библиотека ONLINE.
2. <https://www.java.com/ru/> - Сайт разработчика Java
3. <https://www.rucont.ru/> - Электронная библиотека Руконт.
4. <https://znanium.com/> - Электронная библиотека издательства Инфра-М.
5. <https://e.lanbook.com/> - Электронная библиотека Лань.
6. <https://www.jetbrains.com/ru-ru/idea/> - Сайт разработчика IntelliJ IDEA.
7. <https://github.com/> - Веб-сервис для хостинга IT-проектов и их совместной разработки.
8. <https://www.eclipse.org/> - Сайт разработчика Eclipse
9. <https://jug.ru/> - Сайт Java-разработчиков

## **9. Методические указания для обучающихся по освоению дисциплины**

Методические указания для обучающихся по освоению дисциплины приведены в Приложении 2 к настоящему Положению.

## **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

**Перечень программного обеспечения:**

Microsoft Office или свободно распространяемые аналоги, Double Commander, VS Code, Eclipse IDE, IntelliJ IDEA, JDK Java SE8 и выше, Notepad++.

**Информационные справочные системы:**

1. Электронные ресурсы образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Объектно-ориентированное программирование».

**11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине****Лекционные занятия:**

- аудитория, оснащенная презентационной техникой (проектор, экран);
- комплект электронных презентаций/слайдов.

**Практические занятия:**

- учебный класс, оснащенный вычислительной техникой (ПК) и доступом к Интернет-ресурсам.
- рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
- рабочие места студентов, оснащенные компьютерами с доступом в Интернет.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И  
ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ  
ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО  
ДИСЦИПЛИНЕ  
«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»  
(Приложение 1 к рабочей программе)**

**Направление подготовки: 09.03.04 Программная инженерия**

**Профиль: Проектирование и разработка программного обеспечения**

**Уровень высшего образования: бакалавриат**

**Форма обучения: очная**

## 1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п/п	Индекс компетенции	Содержание компетенции (или ее части)*	Раздел дисциплины, обеспечивающий формирование компетенции (или ее части)	В результате изучения раздела дисциплины, обеспечивающего формирование компетенции (или ее части), обучающийся должен:		
				Необходимые знания	Необходимые умения	Трудовые действия
1	ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	Тема 1 Тема 2	Знает принципы работы современных информационных технологий и программных средств, в том числе отечественного производства.	Анализирует современные информационные технологии и программные средства, в том числе, отечественного производства при решении задач профессиональной деятельности;  Использует современные информационные технологии и программные средства, в том числе, отечественного производства, при решении задач профессиональной деятельности;	Применяет современные информационные технологии и программные средства, в том числе, отечественного производства при решении задач профессиональной деятельности;
2	ОПК-3	Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Тема 1 Тема 2	Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом	Использует знания об основах информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности для решения стандартных задач профессиональной деятельности;	Применяет принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом

				основных требований информационной безопасности		основных требований информационной безопасности;  Анализирует информационно-коммуникационных технологий и с учетом основных требований информационной безопасности для подготовки обзоров, аннотаций, составления рефератов, научных докладов;
3.	ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов.	Тема 3 Тема 4 Тема 5	Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий	Анализирует современные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды для разработки информационных систем и технологий;  Использует современные языки программирования, программные среды разработки информационных систем и технологий для отладки и тестирования прототипов программно-технических комплексов задач;	Применяет основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий;
4.	ОПК-7	Способен применять в практической деятельности	Тема 6	Знает основные концепции, принципы, теории и	Использует основные концепции, принципы,	Применяет в практической деятельности основные

	основные концепции, принципы, теории и факты, связанные с информатикой.		факты, связанные с информатикой	теории и факты, связанные с информатикой для решения практических задач;	концепции, принципы, теории и факты, связанные с информатикой;
--	---	--	---------------------------------	--	--

## 2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Код компетенции	Инструмент, оценивающий сформированность компетенции	Показатель оценивания компетенции	Критерии оценки
ОПК-2, ОПК-3, ОПК-6, ОПК-7	Контрольная работа	<p>А) полностью сформирована (компетенция освоена на <u>высоком</u> уровне) – 5 баллов</p> <p>Б) частично сформирована:</p> <ul style="list-style-type: none"> <li>• компетенция освоена на <u>продвинутом</u> уровне – 4 балла;</li> <li>• компетенция освоена на <u>базовом</u> уровне – 3 балла;</li> </ul> <p>В) не сформирована (компетенция <u>не сформирована</u>) – 2 и менее баллов</p>	<p>1. Проводится в форме письменной работы</p> <p>2. Время, отведенное на процедуру – семестр.</p> <p>Неявка на защиту контрольной работы – 0.</p> <p>Критерии оценки:</p> <p>1. Соответствие содержания контрольной работы заявленной тематике (1 балл).</p> <p>2. Качество источников и их количество при подготовке работы (1 балл).</p> <p>3. Владение информацией и способность отвечать на вопросы аудитории (1 балл).</p> <p>4. Качество самой представленной работы (1 балл).</p> <p>5. Использование специализированного программного обеспечения (1 балл).</p> <p>6. Оригинальность подхода и всестороннее раскрытие выбранной тематики (1 балл).</p>

			<p>Максимальная сумма баллов - 6 баллов.</p> <p>Результаты оценочной процедуры представляются обучающимся в срок не позднее 1 недели после проведения процедуры – для текущего контроля.</p> <p>Оценка проставляется в электронный журнал.</p>
--	--	--	--

### **3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

#### **3.1. Примерная тематика заданий на контрольную работу:**

**1.** Реализовать приложение, вычисляющее для последовательности чисел, представляющих величины углов, следующие тригонометрические функции:

- сумма синусов/косинусов значений,
- разность синусов/косинусов значений,
- произведение синусов/косинусов введенных значений.

Единица измерения углов должна указываться как параметр командной строки. Для представления единиц измерения и используйте константы с типом `int`. Для хранения значений, необходимо использовать массив с типом `double`.

Объявления методов должны выглядеть следующим образом:

```
/**
```

```
@param angles – массив значений углов;
```

```
@param type - единица измерения, имеющее значение одной из констант
```

```
@return сумму синусов значений углов;
```

```
*/
```

```
public static double
```

```
getSumSinuses(double[] angles,int type);
```

**2.** С помощью перегрузки реализуйте методы, позволяющие указывать количество значащих знаков для результатов выполнения операций.

**3.** Дополнить приложение, таким образом, чтобы имелась возможность формирования исходной числовой последовательности числами, сформированными генератором случайных чисел;

**4.** Реализовать класс объявляемой исключительной ситуации, для проверки корректности вводимых данных. Объект данной исключительной

ситуации содержать в себе информацию о значении, обработка которого вызвала данную исключительную ситуацию

5. Реализовать класс, в котором будут содержаться все описанные выше функциональные возможности

#### 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Формой контроля знаний по дисциплине «Объектно-ориентированное программирование» являются две текущие аттестации в форме тестов и итоговая аттестация в форме зачета с оценкой.

Неделя текущей/промежуточной контрольной	Вид оценочного средства	Код компетенций, оцениваемый знания, умения, навыки	Содержание оценочного средства	Требования к выполнению	Срок сдачи (неделя семестра)	Критерии оценки по содержанию и качеству с указанием баллов
7-8 15-16	Тестирование	ОПК-2, ОПК-3, ОПК-6, ОПК-7	25 вопросов	Компьютерное тестирование. Время отведенное на процедуру – 30 минут	Результаты тестирования предоставляются в день проведения процедуры	Критерии оценки определяются процентным соотношением. Не явка - 0 Удовлетворительно - от 51% правильных ответов. Хорошо - от 70%. Отлично – от 90%.
В соответствии с КУГ	Зачёт с оценкой	ОПК-2, ОПК-3, ОПК-6, ОПК-7	2 вопроса	Зачет с оценкой проводится в письменной форме, путем ответа на вопросы. Время, отведенное на процедуру – 30 минут.	Результаты предоставляются в день проведения экзамена	Критерии оценки: « <b>Отлично</b> »: <ul style="list-style-type: none"> <li>• знание основных понятий предмета;</li> <li>• умение использовать и применять полученные знания на практике;</li> <li>• работа на практических занятиях;</li> <li>• знание основных научных теорий, изучаемых предметов;</li> <li>• ответ на вопросы билета.</li> </ul> « <b>Хорошо</b> »: <ul style="list-style-type: none"> <li>• знание основных понятий предмета;</li> <li>• умение использовать и применять полученные знания на практике;</li> <li>• работа на практических занятиях;</li> <li>• знание основных научных теорий, изучаемых предметов;</li> <li>• ответы на вопросы билета неполные</li> </ul> « <b>Удовлетворительно</b> »: <ul style="list-style-type: none"> <li>• демонстрирует частичные знания по темам дисциплины;</li> <li>• незнание неумение использовать и применять</li> </ul>

						<p>полученные знания на практике;</p> <ul style="list-style-type: none"> <li>• не работал на практических занятиях;</li> </ul> <p>«Неудовлетворительно»:</p> <ul style="list-style-type: none"> <li>• демонстрирует частичные знания по темам дисциплин;</li> <li>• незнание основных понятий предмета;</li> <li>• неумение использовать и применять полученные знания на практике;</li> <li>• не работал на практических занятиях;</li> </ul> <p>не отвечает на вопросы.</p>
--	--	--	--	--	--	---

#### 4.1. Типовые вопросы, выносимые на тестирование:

##### 1. Объектно-ориентированный подход к проектированию программных продуктов основан на следующих принципах:

- 1) выделение классов объектов;
- 2) установление характерных свойств объектов и методов их обработки
- 3) создание иерархии классов, наследовании свойств объектов и методов их обработки.
- 4) Ничего из вышеперечисленного

##### 2. Структурный подход использует:

- 1) диаграммы потоков данных (информационно-технологические схемы) – показывают процессы и информационные потоки между ними с учетом "событий", инициирующих процессы обработки;
- 2) диаграммы декомпозиции – структура и декомпозиция целей, функций управления, приложений
- 3) структурные схемы – архитектура программного продукта в виде иерархии взаимосвязанных программных модулей с идентификацией связей между ними, детальная логика обработки данных программных модулей (блок-схемы).
- 4) Ничего из выше перечисленного.

##### 3. Прародителями всех языков ООП является

- 1) Симула
- 2) Delphi
- 3) C#
- 4) Объектом объектно-ориентированного программирования называется ...

##### 4. Совокупность переменных состояния и связанных с ними методов (операций)

- 1) группа данных и методов (функций) для работы с этими данными
- 2) функция или процедура, выполняющие определенные действия
- 3) характеристика, назначенная элементу класса

##### 5. Выберите наиболее подходящее определение Класса.

- 1) Тип, содержащий набор функций
- 2) Тип, который отображает состояние некоторого объекта
- 3) Тип, описывающий поведение некоторой сущности
- 4) Тип, описывающий характеристики и поведение объекта

**6. Двумерным называется массив, элементы которого расположены в виде...**

- 1) множества измерений
- 2) квадратной таблицы
- 3) прямоугольной таблицы

**7. Строка `Readln (a [ i, j ] )`**

- 1) ожидает ввод размера массива
- 2) ожидает ввод элемента массива
- 3) ожидает вывод массива

**8. Выберите язык ООП**

- 1) Borland Delphi
- 2) Pascal
- 3) Basic

**9. Любое условие в программном коде записывается**

- 1) IF...THEN...ELSE
- 2) Begin...END
- 3) FOR..TO..DO

**10. В каком разделе задаются константы?**

- 1) В var
- 2) после begin
- 3) перед var

**11. Как записывается цикл в программном коде?**

- 1) IF...THEN...ELSE
- 2) Begin...END
- 3) FOR..TO..DO

**12. Выберите правильную форму записи в var?**

- 1) имя\_переменной:тип;
- 2) тип:имя\_переменной;

**13. Для расширения возможностей можно...**

- 1) Подключить дополнительные библиотеки
- 2) Создать дополнительный цикл

**14. Любой язык программирования переводит свой код в ...**

- 1) машинный язык
- 2) JavaScript
- 3) Остается неизменным

**15. Какие языки программирования набирают популярность**

- 1) ООП
- 2) Консольные

**16. Можно ли сказать, что Delphi является универсальным языком программирования?**

- 1) Да
- 2) Нет
- 3) Это не язык программирования

**17. Для исправления ошибок в исходном коде, во время компиляции используется**

- 1) декомпилятор
- 2) дебагер

**18. Для правильного сохранения проекта следует**

- 1) Файл-Сохранить
- 2) Файл-Сохранить Как
- 3) Файл-Сохранить проект Как

**19. Для чего предназначена технология программирования?**

- 1) Для решения вычислительных обязанностей и финансовых служб
- 2) Для обработки точного изображения и звука
- 3) Для обработки больших структурированных объемов информации
- 4) Для создания инструментальных программных инструментов информационных технологий
- 5) Для обработки текстовой информации

**20. Укажите, какое расширение имеет файл проекта**

- 1) .prj
- 2) .dcs
- 3) .pas
- 4) .exe
- 5) .dfm

**21. Для изменения значений свойств компонентов в процессе разработки программы используется:**

- 1) Property Editor
- 2) Image Editor
- 3) Object Inspector
- 4) Code Editor
- 5) Project Options

**22. Основные принципы ООП -**

- 1) Инкапсуляция, наследование, полиморфизм
- 2) Наследование, дедукция, инкапсуляция
- 3) Полиграфия, делегирование, наследование
- 4) Индукция, наследование, полиморфизм
- 5) Инкапсуляция, наследование, делегирование

**23. Выберите последовательность областей видимости, упорядоченную по возрастанию видимости методов.**

- 1) Protected, public, private
- 2) Public, protected, private
- 3) Private, protected, public k
- 4) Protected, public, private
- 5) Public, private, protected

**24. Выберите выражение, в результате которого получено значение  $c=3$ , если  $a=14$  и  $b=4$**

- 1)  $c:=b \text{ mod } a$
- 2)  $c:= a \text{ mod } b$
- 3)  $c:=a/b$
- 4)  $c:=b \text{ div } a$
- 5)  $c:= a \text{ div } b$

**25. Укажите компонент Контейнер**

- 1) PageControl

- 2) StatusBar
- 3) Memo
- 4) Panel
- 5) Edit

**26. Возможность использовать одинаковые имена для методов, входящих различные классы называется...**

- 1) Наследование
- 2) Метоморфизм
- 3) Декапсуляция
- 4) Инкапсуляция
- 5) Полиморфизм

**27. Конструкторы...**

- 1) предназначены для создания таблицы виртуальных методов
- 2) являются статическими методами
- 3) служат для возможности наследования виртуальных методов или их переопределения
- 4) они присутствуют во всех объектных типах (классах) и предназначены для создания экземпляров объектов
- 5) выполняются обязательно перед первым использованием виртуальных методов

**28. Укажите сколько файлов будет создано при сохранении проекта, в состав которого входит только одна форма**

- 1) 7
- 2) 6
- 3) 4
- 4) 5
- 5) 2

**29. Массивы бывают**

- 1) Числовые и вещественные
- 2) Одномерные и многомерные
- 3) Одномерные и разномерные
- 4) Строчные и числовые
- 5) Символьные или файловые

**30. Основные операции с массивами - это**

- 1) Перебор элементов массива, нахождение номера индекса
- 2) Подсчет индекса, количества элементов
- 3) Суммирование индексов, ввод значений в элементы массива
- 4) Определение типа элемента массива, сравнение элементов массива
- 5) Ввод и вывод значений в элементы массива, поиск максимального или минимального элемента, нахождение суммы элементов массива

**31. Укажите различие функций от процедуры:**

- 1) В описании функций указываются списки формальных параметров
- 2) Функция возвращает значение, а при вызове функций в программе, используется как переменная в выражении
- 3) В функции описываются все метки, константы, типы и переменные
- 4) Нет различия

5) Можно вызвать функцию в любом разделе главной программы

### **32. Что такое алгоритм**

1) Алгоритм - это совокупность требований к программе

2) Алгоритм - это индивидуальные варианты исполнителю для выполнения какого-либо действия.

3) Алгоритм - это порядок команд для ЭВМ.

4) Алгоритм - это совокупность справочников и правил к исполнителю.

5) Алгоритм - это точное и понятное предписание, определяющее процесс перехода от исходных данных к результату.

### **33. Формальные параметры делятся на ...**

1) Значения, переменные, константы

2) Константы, локальные

3) Локальные, глобальные

4) Переменные, глобальные

5) Фактические, глобальные

### **34. Блок-схема это...**

1) Способ представления алгоритма

2) Вид алгоритма

3) Решение задачи

4) Программа

5) Команда алгоритма

### **35. Массив – это...**

1) нефиксированная структура данных, содержащих переменные одного типа данных;

2) нефиксированная структура данных, содержащих переменные разного типа данных;

3) фиксированный набор данных, имеющий общее имя, содержащий переменные разного типа данных

4) структура данных, содержащих переменные одинакового типа и имеющих одинаковые значения.

5) упорядоченная последовательность однотипных данных, расположенных на носителе информации

### **36. Последовательность символов представляет собой**

1) пользовательский тип данных

2) простой тип данных

3) символьный тип данных

4) скалярный тип данных

5) строковый тип данных

### **37. Слова, смысл и способ употребления которых задан раз и навсегда называют:**

1) Серией

2) Составными словами

3) Командами

4) Служебными словами

5) Простыми словами

### **38. Свойство алгоритма «дискретность» означает:**

- 1) являются слова или тексты
- 2) Способность алгоритма давать правильные результаты решения задач
- 3) Предлагаемые действия должны быть понятными и единственно возможными
- 4) Пригодность алгоритма для решения однотипных задач
- 5) Алгоритм состоит из отдельных шагов. Каждый шаг алгоритма – это некоторое законченное действие.

**39. Линейным называется алгоритм, при котором...**

- 1) последовательность действий совершается одна за другой в порядке их следования;
- 2) Используется оператор цикла
- 3) Используется оператор безусловного перехода
- 4) Используется оператор FOR
- 5) Используется оператор GOTO

**40. Выберите идентификатор, который используется для обозначения символьного типа данных**

- 1) varchar
- 2) string
- 3) symbol
- 4) char
- 5) comp

#### **4.2. Типовые вопросы, выносимые на зачёт с оценкой**

- 1) Что такое объектно-ориентированная парадигма?
- 2) Какие основные понятия включает объектно-ориентированное программирование?
- 3) Каковы преимущества и недостатки объектно-ориентированного программирования?
- 4) Как создать класс и объект в Python?
- 5) Что такое методы и атрибуты классов?
- 6) Какие методы выполняются автоматически при создании и удалении объекта?
- 7) Представьте практический пример использования классов и объектов.
- 8) Как применяется принцип наследования в объектно-ориентированном программировании?
- 9) Что такое переопределение методов и атрибутов?
- 10) Как создать абстрактный класс и интерфейс?
- 11) Что такое полиморфизм и как он реализуется?
- 12) Какие механизмы обеспечивают инкапсуляцию в языке программирования?
- 13) Что такое геттеры и сеттеры, и зачем они используются?
- 14) Приведите примеры реализации инкапсуляции.
- 15) Какие механизмы обработки ошибок и исключений существуют?
- 16) Объясните принцип единственной ответственности (SRP).

- 17) Как принцип открытости/закрытости (ОСР) влияет на расширяемость кода?
- 18) Что означает принцип подстановки Барбары Лисков (LSP)?
- 19) Какой смысл принципа разделения интерфейса (ISP)?
- 20) В чем заключается принцип инверсии зависимостей (DIP)?
- 21) Какие методы позволяют работать с файлами и потоками данных?
- 22) Какие основные исключения могут возникнуть при работе с файлами?
- 23) Как обрабатываются исключения в Python?
- 24) Что такое паттерны проектирования и для чего они используются?
- 25) Какой паттерн проектирования реализует создание единственного экземпляра класса?
- 26) Что делает паттерн "Фабричный метод"?
- 27) Как работает паттерн "Адаптер" и в каких ситуациях он используется?
- 28) Что такое паттерн "Декоратор" и как он расширяет функциональность объекта?
- 29) Как паттерн "Наблюдатель" позволяет реализовать обратную связь между объектами?
- 30) Какой паттерн позволяет выбирать алгоритм выполнения во время выполнения программы?
- 31) Что такое ORM и какие преимущества он предоставляет?
- 32) Какие основные операции можно выполнять с данными в базе данных?
- 33) Каким образом происходит создание и подключение к базе данных в Python?
- 34) Приведите пример работы с базой данных с использованием языка SQL.
- 35) В чем заключается юнит-тестирование?
- 36) Какие принципы написания хороших тестов соблюдаются при тестировании классов и методов?
- 37) Какие виды ошибок можно обнаружить с помощью инструментов отладки?
- 38) Какие инструменты используются для поиска ошибок в программном коде?

Итоговое начисление баллов по дисциплине осуществляется в соответствии с разработанной и внедренной балльно-рейтинговой системой контроля и оценивания уровня знаний и внеучебной созидательной активности обучающихся.

**ИНСТИТУТ ИНФОКОММУНИКАЦИОННЫХ СИСТЕМ И  
ТЕХНОЛОГИЙ  
КАФЕДРА ПРИКЛАДНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО  
ОСВОЕНИЮ ДИСЦИПЛИНЫ  
«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

**(Приложение 2 к рабочей программе)**

**Направление подготовки: 09.03.04 Программная инженерия**

**Профиль: проектирование и разработка программного обеспечения**

**Уровень высшего образования: бакалавриат**

**Форма обучения: очная**

## **1. Общие положения**

Целью дисциплины является формирование у обучающихся системного представления об особенностях создания программ с помощью объектно-ориентированного подхода и формирования практических навыков по созданию, отладке и модификации прикладных программ, написанных с использованием паттернов объектно-ориентированного программирования.

Основными **задачами** дисциплины является:

- ознакомление с основными понятиями и терминологией паттернов объектно-ориентированного программирования;
- ознакомление с областями применения объектно-ориентированного программирования;
- ознакомление с концепцией объектно-ориентированного подхода применительно к бизнес-деятельности;
- выработка практических навыков в области выбора и применения паттернов объектно-ориентированного программирования.

## **2. Указания по проведению практических (семинарских) занятий**

### **Практическая работа 1.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 1. Основы языка объектно-ориентированного программирования Java.

- Лексические основы языка Java.
- Управляющие конструкции языка Java.
- Ввод-вывод в Java.

Продолжительность занятия 4 часа.

### **Практическая работа 2.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 2. Реализация концепций объектно-ориентированного программирования в Java.

- Создание класса Java
- Тип String и операторы
- Тип StringBuffer (StringBuilder)

Продолжительность занятия 4 часа

### **Практическая работа 3.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 2. Реализация концепций объектно-ориентированного программирования в Java.

- Массивы
- Коллекции Java
- Архивирование Java-кода

Продолжительность занятия 4 часа.

#### **Практическая работа 4.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 3. Расширенные представления объектно-ориентированного подхода.

- Перегружаемые методы

Продолжительность занятия 4 часа.

#### **Практическая работа 5.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 3. Расширенные представления объектно-ориентированного подхода.

- Сравнение объектов.
- Исключения.

Продолжительность занятия 4 часа.

#### **Практическая работа 6.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Создание Java-приложений.

- Интерфейсы.
- Программирование ввода/вывода

Продолжительность занятия 4 часа

#### **Практическая работа 7.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Создание Java-приложений.

- Вложенные классы
- Тип Generic

Продолжительность занятия 4 часа.

#### **Практическая работа 8.**

Вид практического занятия: компьютерное моделирование.

Образовательные технологии: технологии компьютерного обучения.

Тема и содержание занятия: Тема 4. Создание Java-приложений.

- Сериализация Java

Продолжительность занятия 4 часа.

### 3. Указания по проведению лабораторного практикума

Не предусмотрено учебным планом

### 4. Указания по проведению самостоятельной работы студентов

№ п/п	Наименование блока (раздела) дисциплины	Виды СРС
1.	Тема 1. Введение в паттерны проектирования	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (разработка собственных компонентов и их установка в систему).
2.	Тема 2. Проектирование редактора документов	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (разработка собственных компонентов и их установка в систему).
3.	Тема 3. Порождающие паттерны	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (разработка собственных компонентов и их установка в систему).
4.	Тема 4. Структурные паттерны	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (создание собственных библиотек).
5.	Тема 5. Паттерны поведения	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (создание собственных библиотек).
6.	Тема 6. Реализация паттернов проектирования в языках высокого уровня	1. Подготовка к практическим занятиям по материалам лекций и учебной литературы. 2. Выполнение практических заданий 3. Самостоятельное изучение некоторых вопросов дисциплины (создание собственных библиотек).

### 5. Перечень основной и дополнительной учебной литературы

#### Основная литература:

1. Барков, И. А. Объектно-ориентированное программирование / И. А. Барков. — 2-е изд., стер. — Санкт-Петербург: Лань, 2023. — 700 с. — ISBN 978-5-507-47113-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/329549>

2. Унгер, А. Ю. Объектно-ориентированное программирование: учебное пособие / А. Ю. Унгер. — Москва: РТУ МИРЭА, 2022. — 84 с. — ISBN 978-5-7339-1628-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/265691>

3. Кувшинов, Д. Р. Основы программирования: учебное пособие для вузов / Д. Р. Кувшинов. — Москва: Издательство Юрайт, 2023. — 104 с. — (Высшее образование). — ISBN 978-5-534-07559-5. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/533784>

4. Бабушкина И. А. Практикум по объектно-ориентированному программированию: учебное пособие / И. А. Бабушкина, С. М. Окулов. — 5-е изд. — Москва: Лаборатория знаний, 2020. — 369 с. — ISBN 978-5-00101-780-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/135561>

5. Зайцев М. Г. Объектно-ориентированный анализ и программирование: учебное пособие: [16+] / М. Г. Зайцев; Новосибирский государственный технический университет. — Новосибирск: Новосибирский государственный технический университет, 2017. — 84 с.: ил., табл. — Режим доступа: по подписке. — URL: <https://biblioclub.ru/index.php?page=book&id=576800>

### **Дополнительная литература:**

1. Объектно-ориентированное программирование на C++: учебник / И. В. Баранова, С. Н. Баранов, И. В. Баженова [и др.]. — Красноярск: СФУ, 2019. — 288 с. — ISBN 978-5-7638-4034-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/157572>

2. Рик Г. Объектно-Ориентированное Программирование / Г. Рик; под редакцией Н.Комлева. — Москва: СОЛОН-Пресс, 2018. — 298 с. — ISBN 978-5-91359-285-9. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/107669>

3. Чернышев С. А. Основы программирования на Python: учебное пособие для среднего профессионального образования / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2023. — 349 с. — (Профессиональное образование). — ISBN 978-5-534-17056-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/532292>

### **6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

#### **Интернет-ресурсы:**

1. <https://biblioclub.ru/> - Университетская библиотека ONLINE.
2. <https://www.java.com/ru/> - Сайт разработчика Java
3. <https://www.rucont.ru/> - Электронная библиотека Руконт.
4. <https://znanium.com/> - Электронная библиотека издательства Инфра-М.
5. <https://e.lanbook.com/> - Электронная библиотека Лань.
6. <https://www.jetbrains.com/ru-ru/idea/> - Сайт разработчика IntelliJ IDEA.
7. <https://github.com/> - Веб-сервис для хостинга IT-проектов и их совместной разработки.
8. <https://www.eclipse.org/> - Сайт разработчика Eclipse
9. <https://jug.ru/> - Сайт Java-разработчиков

**Перечень программного обеспечения:**

Microsoft Office или свободно распространяемые аналоги, Double Commander, VS Code, Eclipse IDE, IntelliJ IDEA, JDK Java SE8 и выше, Notepad++.

**Информационные справочные системы:**

1. Электронные ресурсы образовательной среды Университета.
2. Рабочая программа и методическое обеспечение по дисциплине «Объектно-ориентированное программирование».