МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Колледж информационных и креативных технологий

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

МДК.01.01 Разработка программных модулей

ПМ.01 Разработка программных модулей

программы подготовки специалистов среднего звена
09.02.10 Разработка компьютерных игр, дополненной и виртуальной реальности

Форма обучения: очная

Владивосток 2026

Рабочая программа учебной дисциплины МДК.01.01 Разработка программных можулей разработана в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.10 Разработка компьютерных игр, дополненной и виртуальной реальности, утвержденного приказом Минобрнауки России от 25 июня 2024, № 441, примерной образовательной программой.

Разработчик(и): Н.В. Кравцова, преподаватель

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № _9_ от «_20_» ___мая____2026_ г.

Председатель ПЦК _____ Н.В. Кравцова
подпись

СОДЕРЖАНИЕ

1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	**4**
2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	**6**
3 УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	**9**
4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	**11**

1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

1.1 Место дисциплины в структуре основной образовательной программы

Учебная дисциплина МДК.01.01 Разработка программных модулей является частью профессионального цикла ПМ.01 Разработка программных модулей основной образовательной программы (далее ООП) в соответствии с ФГОС СПО по специальности/профессии 09.02.10 Разработка компьютерных игр, дополненной и виртуальной реальности.

Цель: формирование у обучающихся профессиональных компетенций, необходимых для проектирования, разработки, тестирования и документирования программных модулей, предназначенных для реализации алгоритмов машинного обучения и других компонентов систем искусственного интеллекта.

1.2. Планируемые результаты освоения междисциплинарного курса

По итогам освоения курса, обучающиеся должны продемонстрировать результаты обучения, соотнесённые с результатами освоения ООП СПО, приведенные в таблице.

Код	Наименование видов деятельности и профессиональных компетенций
ПК 1.1	Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
ПК 1.2	Разрабатывать программные модули в соответствии с техническим заданием.
ПК 1.3	Оформлять программный код в соответствии с техническим заданием.
ПК 1.4	Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки.

В результате освоения междисциплинарного курса обучающийся должен:

Код ПК	Уметь	Знать	Владеть навыками
ПК 1.1	Анализировать технические задания и выявлять требования к алгоритмам. Применять методы алгоритмизации для решения задач программирования. Разрабатывать оптимальные алгоритмы для решения задач в области ИИ.	Основные методы и подходы к построению алгоритмов (типовые поисковые алгоритмы, жадные алгоритмы, динамическое программирование, рекурсивные подходы). Принципы эффективной обработки данных. Языки программирования, применяемые для разработки алгоритмов.	Разработки, оптимизации и оценки сложности алгоритмов для ИИ-программ. Использования библиотек и инструментов для работы с алгоритмами и данными (например: Pandas, NumPy, Scikit-learn). Применения структур данных (деревья, графы, списки) для реализации алгоритмов.
ПК 1.2	Реализовывать программные модули на	Принципы модульного	Разработки модульных ИИ-систем, соответствующих

	основе требований технического задания. Соблюдать при разработке принципы «чистого кода». Использовать стандартные библиотеки и фреймворки для ускорения разработки.	программирования. Языки программирования для разработки модулей. Стандартные фреймворки и библиотеки для работы с ИИ.	требованиям производительности и безопасности. Внедрения разработанных ИИ-модулей в комплексные программные системы. Оптимизации кода и работы с интерфейсами для взаимодействия между модулями.
ПК 1.3	Оформлять код в соответствии с принятыми стандартами и требованиями. Документировать разработанный программный код. Соблюдать соглашения о наименованиях переменных, функций и классов (например, PEP8 для Python).	Основные принципы чистого кода (Clean Code). Стандарты и практики документирования программного обеспечения. Инструменты для автоматической проверки качества кода (например, PyLint, ESLint).	Оформления, документирования и структурирования кода для последующей поддержки. Использования инструментов статического анализа кода для выявления ошибок и улучшения качества. Работы с системами документирования кода (например, Doxygen, Sphinx).
ПК 1.4	Работать с системами контроля версий для управления проектами. Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода.	Принципы работы распределенных систем контроля версий. Основные команды и операции в системах контроля версий (например: commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки.	Управления проектами с использованием систем контроля версий для организации командной работы. Разрешения конфликтов при слиянии веток и использования pull request для рецензирования кода. Настройки процессов CI/CD для автоматического тестирования и развертывания кода.

2 СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1 Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Объем образовательной программы учебной дисциплины	**36**
в том числе:	
– теоретическое обучение	14
– практические занятия	14
– лабораторные занятия	
– курсовая работа (проект)	
– самостоятельная работа	8
– консультации	
– промежуточная аттестация – диффер. зачет	5

2.2 Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала и формы организации деятельности обучающихся	Объем в часах	Коды компетенций, формированию которых способствует элемент программы
1	2	3	4
Раздел 1. Основы разработки программных модулей		**18**	
Тема 1.1. Введение в разработку программных модулей	**Содержание учебного материала** Понятие программного модуля. Принципы модульного программирования. Роль модулей в разработке игр и VR/AR-приложений.	2	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
Тема 1.2. Жизненный цикл программного модуля	**Содержание учебного материала** Этапы разработки: анализ, проектирование, реализация, тестирование, сопровождение.	4	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
Тема 1.3. Инструменты разработки программного обеспечения	**Содержание учебного материала** Среды разработки (IDE), системы контроля версий, средства сборки проектов. **Самостоятельная работа обучающихся** Провести обзор одной из IDE (Visual Studio, Visual Studio Code, JetBrains Rider и др.). Описать основные функции и возможности.	6	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
Тема 1.4. Основы структурного и объектно-ориентированного программирования	**Содержание учебного материала** Классы, объекты, методы, инкапсуляция, наследование, полиморфизм.	6	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
Раздел 2. Практическая разработка программных модулей для игр и VR/AR		**18**	
Тема 2.1.	**Содержание учебного материала**	4	

Проектирование игровых программных модулей	Архитектура игровых приложений, выделение модулей (логика, физика, UI, управление).		ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
Тема 2.2. Разработка игровых механик в программных модулях	**Содержание учебного материала**	2	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
	Реализация движения, взаимодействия объектов, событий и триггеров.		
	В том числе практических занятий	4	
	Практическое занятие направлено на реализацию базовых игровых механик, таких как движение персонажа, взаимодействие с объектами и обработка пользовательского ввода. Обучающиеся создают простой игровой модуль с использованием событий и триггеров.		
Тема 2.3. Интеграция программных модулей в игровые движки	**Содержание учебного материала**	4	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
	Работа с Unreal Engine, подключение скриптов и компонентов. Практическое занятие включает подключение и настройку скриптов в игровом движке (Unity или Unreal Engine). Студенты закрепляют навыки интеграции модулей в готовую сцену и проверяют их корректную работу.		
Тема 2.4. Тестирование и оптимизация программных модулей	**Содержание учебного материала**	2	ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4
	Отладка, профилирование производительности, исправление ошибок и оптимизация под VR/AR.		
	В том числе практических занятий	2	
	Практическое занятие направлено на выявление и исправление ошибок в программных модулях с использованием инструментов отладки. Также выполняется базовая оптимизация производительности для корректной работы в игровых и VR/AR-средах.		
Промежуточная аттестация		5	
Всего:		**36**	

3 УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1 Материально-техническое обеспечение

Для реализации программы предусмотрено наличие следующих помещений:

Лаборатория организации и принципов построения информационных систем.

Основное оборудование: Компьютеризированные рабочие места обучающихся с выходом в сеть "Интернет" - 20 шт., стулья - 20 шт., комплект мебели (стол и стул) для преподавателя - 1 шт., графические станции ПК №2 WorkStation WS 917 20 шт., автоматизированное рабочее место преподавателя ПК №2 WorkStation WS 917, графический планшет Wacom - 20 шт., мультимедийное оборудование - 1 шт., телевизор, экран, многофункциональное устройство (МФУ) формат А3 -1 шт., доска маркерная - 1 шт.

Программное обеспечение: PyCharm, Visual Studio Code, IntelliJ IDEA, JetBrains Rider, Python (Anaconda3) с TensorFlow/PyTorch, Microsoft Visio, Draw.io, Figma, Figma, Adobe Illustrator, MySQL Workbench, HeidiSQL, MySQL, PostgreSQL, Git, Docker Desktop, Microsoft Project, Draw.io, Microsoft SQL Server Management Studio, Microsoft Office.

Помещение для самостоятельной и воспитательной работы обучающихся, оснащенное компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации.

Основное оборудование: Рабочие места на базе компьютерной техники с возможностью подключения к информационно-телекоммуникационной сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду ВВГУ: комплекты учебной мебели (столы и стулья) – 19 шт., персональные компьютеры (облачные мониторы) - 19 шт; доска маркерная - 1шт., лазерный копир-принтер-сканер с 2 лотками Xerox WorkCentre 3345 DNI.

Программное обеспечение: Microsoft Windows 7 Professional; Microsoft Office ProPlus 2010 Russian Acdmc; СПС КонсультантЮрист: Версия Проф; Adobe Acrobat Reader; Google Chrome; Adobe Flash Player; 7-Zip 18.01 (x64).

3.2 Информационное обеспечение реализации программы

Для реализации программы учебной дисциплины библиотечный фонд ВВГУ укомплектован печатными и электронными изданиями.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

Основные источники

1. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 273 с. — (Профессиональное образование). — ISBN 978-5-534-20362-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/562355

2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 404 с. — (Профессиональное образование).
— ISBN 978-5-534-19506-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/566739

3. Соколова, В. В. Разработка мобильных приложений : учебное пособие для среднего профессионального образования / В. В. Соколова. — Москва : Издательство Юрайт, 2024. —160 с. — (Профессиональное образование). — ISBN 978-5-534-16868-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/542342

Дополнительные источники

1. Черпаков, И. В. Основы программирования : учебник и практикум для среднего профессионального образования / И. В. Черпаков. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 196 с. — (Профессиональное образование). — ISBN 978-5-534- 18760-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/561922

2. Зыков, С. В. Объектно-ориентированное программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 151 с. — (Высшее образование). — ISBN 978-5-534-16941-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/537385

3. Станкевич, Л. А. Интеллектуальные системы и технологии : учебник и практикум для среднего профессионального образования / Л. А. Станкевич. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 478 с. — (Профессиональное образование). — ISBN 978-5-534-20364-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/566524

Электронные ресурсы

1. Информационно-справочная система «Консультант Плюс» – http://www.consultant.ru/
2. Профессиональная база данных: "Открытая база ГОСТов"/ Режим доступа: http://standartgost.ru/, доступ свободный
3. Электронная библиотечная система «РУКОНТ» - Режим доступа: https://lib.rucont.ru/
4. Электронная библиотечная система ZNANIUM.COM - Режим доступа: https://znanium.com/
5. Электронно-библиотечная система издательства "Юрайт" - Режим доступа: https://urait.ru/
6. Электронно-библиотечная система издательства "Лань" - Режим доступа: https://e.lanbook.com/
7. Президентская библиотека им. Б.Н. Ельцина (база данных различных профессиональных областей) Режим доступа: https://www.prlib.ru/, доступ свободный

4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения	Критерии оценки	Методы оценки
Знает принципы модульного и объектно-ориентированного программирования. Знает этапы жизненного цикла программного модуля. Знает инструменты разработки программного обеспечения. Знает основы разработки игровых и VR/AR приложений	Правильно объясняет основные принципы (инкапсуляция, наследование, полиморфизм, модульность), приводит примеры. Перечисляет этапы разработки и описывает их содержание без существенных ошибок. Называет основные IDE, системы контроля версий и средства сборки, объясняет их назначение. Описывает структуру игровых приложений и назначение основных модулей	Тестирование, собеседование письменный опрос устный опрос
Умеет разрабатывать простые программные модули. Умеет создавать игровые механики. Умеет интегрировать модули в игровой движок. Умеет выполнять тестирование и отладку программных модулей	Находит и исправляет ошибки, использует инструменты отладки и профилирования. Реализует корректные модули с использованием базовых конструкций языка программирования. Разрабатывает работающие механики (движение, взаимодействие объектов, события) без критических ошибок. Подключает скрипты и компоненты к сцене и обеспечивает их корректное взаимодействие	Практическая работа, диффер. зачет Лабораторная работа

Для оценки достижения запланированных результатов обучения по дисциплине разработаны контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации, которые прилагаются к рабочей программе дисциплины.

МИНОБРНАУКИ РОССИИ
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Колледж информационных и креативных технологий

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА

для проведения текущего контроля и промежуточной аттестации
по учебной дисциплине

МДК.01.01 Разработка программных модулей

программы подготовки специалистов среднего звена
09.02.10 Разработка компьютерных игр, дополненной и виртуальной реальности

Форма обучения: очная

Владивосток 2026

Контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации по учебной дисциплине МДК.01.01 Разработка программных модулей разработаны в соответствии с требованиями ФГОС СПО по специальности / профессии 09.02.10 Разработка компьютерных игр, дополненной и виртуальной реальности, утвержденного приказом Минобрнауки РФ от от 25 июня 2024, № 441, примерной образовательной программой, рабочей программой учебной дисциплины.

Разработчик(и): Н.В. Кравцова, преподаватель

Рассмотрено и одобрено на заседании цикловой методической комиссии

Протокол № _9_ от «_20_» ___мая____2026_ г.

Председатель ПЦК _____ Н.В. Кравцова

подпись

1 Общие сведения

Контрольно-оценочные средства (далее – КОС) предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины МДК.01.01 Разработка программных модулей.

КОС включают в себя контрольные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине, которая проводится в форме дифференцированного зачёта посредством сдачи индивидуальной аналитико-проектной работы.

2 Планируемые результаты обучения по дисциплине, обеспечивающие результаты освоения образовательной программы

Код ПК	Код результата обучения	Наименование результата обучения
ПК 1.1	З1	Основы алгоритмизации и программирования.
	З2	Принципы модульного и структурного программирования.
	З3	Основные конструкции языка программирования (C#, C++, Python и др.).
	У1	Разрабатывать программные модули по техническому заданию.
	У2	Использовать базовые алгоритмы при создании программ.
	У3	Структурировать код на логические блоки.
ПК 1.2	З4	Виды тестирования программного обеспечения.
	З5	Методы поиска и устранения ошибок.
	З6	Принципы работы отладчиков.
	З7	Основы логирования и обработки ошибок.
	У4	Выявлять ошибки в программном коде.
	У5	Использовать инструменты отладки (debugger).
	У6	Проводить модульное тестирование.
	У7	Анализировать причины возникновения ошибок.
ПК 1.3	З8	Принципы интеграции программных модулей.
	З9	Архитектура программных систем.
	З10	Способы взаимодействия модулей между собой.
	З11	Основы API и интерфейсов взаимодействия.
	У8	Объединять программные модули в единую систему.

Код ПК	Код результата обучения	Наименование результата обучения
	У9	Настраивать взаимодействие между компонентами программы.
	У10	Использовать API для связи модулей.
	У11	Подключать скрипты к игровым сценам.
	У12	Настраивать передачу данных между модулями.
ПК 1.4	З12	Основы оптимизации программного кода.
	З13	Методы повышения производительности программ.
	З14	Причины возникновения ошибок в работающих системах.
	З15	Способы сопровождения программного обеспечения.
	У13	Оптимизировать программный код.
	У14	Улучшать производительность модулей.
	У15	Проводить рефакторинг программного кода.
	У16	Использовать инструменты профилирования.
	У17	Исправлять ошибки в работающих программах.

[1]- *в соответствии с рабочей программой учебной дисциплины*

3 Соответствие оценочных средств контролируемым результатам обучения

3.1 Средства, применяемые для оценки уровня теоретической подготовки

Краткое наименование раздела (модуля) / темы дисциплины	Код результата обучения	Показатель[2] овладения результатами обучения	Наименование оценочного средства и представление его в КОС[3]	
			Текущий контроль[4]	Промежуточная аттестация[4]
Раздел 1. Основы разработки программных модулей				
Тема 1.1. Введение в разработку программных модулей	З1	Способность понимать назначение программных модулей в структуре программного обеспечения.	Устный опрос	Собеседование
	З2	Способность различать виды программных модулей и их функции.		
	У1	Способность объяснять принципы модульного программирования.		
	У2	Способность определять роль модулей в разработке игр и VR/AR приложений.		
	У3	Способность анализировать структуру программного продукта.		
Раздел 2. Практическая разработка программных модулей для игр и VR/AR				

Краткое наименование раздела (модуля) / темы дисциплины	Код результата обучения	Показатель[2] овладения результатами обучения	Наименование оценочного средства и представление его в КОС[3]	
			Текущий контроль[4]	Промежуточная аттестация[4]
Тема 2.1. Проектирование игровых программных модулей	З3	Способность выделять основные компоненты программной системы.	Устный опрос	Собеседование
	З4	Способность приводить примеры использования модулей в практике разработки.		
	У3	Способность формировать представление о процессе разработки программных систем.		
	У4	Способность описывать этапы жизненного цикла программного модуля.		
	У5	Способность различать стадии разработки программного обеспечения.		
	З14	Способность анализировать процессы проектирования и реализации модулей.		
	У16	Способность определять задачи каждого этапа жизненного цикла.		
Тема 2.2. Разработка игровых механик в программных модулях	З4	Способность реализовывать базовые игровые механики.	Лабораторная работа	Собеседование
	З5	Способность программировать движение игровых объектов.		
	З6	Способность обрабатывать пользовательский ввод.		
	У5	Способность создавать взаимодействие объектов в сцене.		
	У11	Способность использовать события и триггеры.		
	У12	Способность разрабатывать простую игровую логику.		
	У13	Способность тестировать игровые механики.		
	У16	Способность модифицировать игровые элементы для улучшения геймплея.		
Тема 2.3. Интеграция программных модулей в игровые движки	З12	Способность подключать программные модули к игровому движку.	Лабораторная работа	Собеседование
	З13	Способность настраивать сцены в Unreal Engine.		
	З14	Способность связывать скрипты с игровыми объектами.		
	У8	Способность обеспечивать взаимодействие компонентов системы.		
	У9	Способность использовать API игрового движка.		
	У10	Способность проверять корректность интеграции модулей.		

Краткое наименование раздела (модуля) / темы дисциплины	Код результата обучения	Показатель[2] овладения результатами обучения	Наименование оценочного средства и представление его в КОС[3]	
			Текущий контроль[4]	Промежуточная аттестация[4]
	У12	Способность устранять ошибки взаимодействия компонентов.		
	У16	Способность адаптировать модули под особенности движка.		
Тема 2.4. Тестирование и оптимизация программных модулей	З9	Способность выявлять ошибки в программных модулях	Практическая презентация	Собеседование
	З10	Способность использовать инструменты отладки.		
	З11	Способность проводить модульное тестирование.		
	У10	Способность анализировать производительность программы.		
	У11	Способность оптимизировать программный код.		
	У15	Способность устранять узкие места в производительности.		

[2] - для формулировки показателей использовать положения Таксономии Блума.
[3] - Однотипные оценочные средства нумеруются, н-р: «Тест №2», «Контрольная работа №4».
[4] - Примеры всех оценочных средств должны быть представлены в разделах 5,6.
[5] - В скобках следует указать пункт разделов 5.6, в котором оно представлено.

3.2 Средства, применяемые для оценки уровня практической подготовки

Краткое наименование раздела (модуля) / темы дисциплины	Код результата обучения	Показатель овладения результатами обучения	Наименование оценочного средства и представление его в КОС	
			Текущий контроль	Промежуточная аттестация
Раздел 1. Основы разработки программных модулей				
Тема 1.2. Жизненный цикл программного модуля	З1	Способность применять знания жизненного цикла при разработке проектов.	Лабораторная работа	Дифф. зачет
	У1	Способность планировать последовательность разработки программного модуля.		
	У2	Способность выявлять ошибки на различных стадиях разработки.		
Тема 1.3. Инструменты разработки программного обеспечения	З2	Способность сравнивать различные программные средства разработки.	Лабораторная работа	Дифф. зачет
	У3	Способность анализировать эффективность используемых инструментов.		
	У4	Способность применять средства сборки программных проектов.		
Тема 1.4. Основы структурного и объектно-	З4	Способность объяснять основные принципы ООП.	Лабораторная работа	Дифф. зачет
	З5	Способность различать классы, объекты и методы.		

Краткое наименование раздела (модуля) / темы дисциплины	Код результата обучения	Показатель овладения результатами обучения	Наименование оценочного средства и представление его в КОС	
			Текущий контроль	Промежуточная аттестация
ориентированного программирования	У5	Способность применять инкапсуляцию в программировании.		
Раздел 2. Практическая разработка программных модулей для игр и VR/AR				
Тема 2.3. Интеграция программных модулей в игровые движки	З13	Способность адаптировать модули под особенности движка.	Лабораторная работа	Дифф. зачет
	У10	Способность устранять ошибки взаимодействия компонентов.		
Тема 2.4. Тестирование и оптимизация программных модулей	З14	Способность проверять корректность интеграции модулей.	Лабораторная работа	Дифф. зачет
	У12	Способность обеспечивать стабильную работу программного продукта.		
	У14	Способность применять методы рефакторинга.		

4 Описание процедуры оценивания

Результаты обучения по дисциплине, уровень сформированности компетенций оцениваются по четырёх бальной шкале оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» (по бальной системе. Максимальная сумма баллов по дисциплине равна 100 баллам.)

Текущая аттестация по дисциплине проводится с целью систематической проверки достижений обучающихся. Объектами оценивания являются: степень усвоения теоретических знаний, уровень овладения практическими умениями и навыками по всем видам учебной работы, качество выполнения самостоятельной работы, учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость всех видов занятий по аттестуемой дисциплине).

При проведении промежуточной аттестации оценивается достижение студентом запланированных по дисциплине результатов обучения, обеспечивающих результаты освоения образовательной программы в целом. Оценка выставляется с учетом оценок, полученных при прохождении текущей аттестации.

Критерии оценивания устного ответа

(оценочные средства: устный опрос, устный доклад, собеседование)

5 баллов - ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

4 балла - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

3 балла – ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

2 балла – ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; незнание современной проблематики изучаемой области.

Критерии оценивания письменной работы

(оценочные средства: индивидуальный проект, практическая презентация)

5 баллов - студент выразил своё мнение по сформулированной проблеме, аргументировал его, точно определив ее содержание и составляющие. Проблема раскрыта полностью, выводы обоснованы. Приведены данные отечественной и зарубежной литературы, статистические сведения, информация нормативно-правового характера. Студент владеет навыком самостоятельной работы по заданной теме; методами и приемами анализа теоретических и/или практических аспектов изучаемой области. Фактических ошибок, связанных с пониманием проблемы, нет; графически работа оформлена правильно.

4 балла - работа характеризуется смысловой цельностью, связностью и последовательностью изложения; допущено не более 1 ошибки при объяснении смысла или содержания проблемы. Проблема раскрыта. Не все выводы сделаны и/или обоснованы. Для аргументации приводятся данные отечественных и зарубежных авторов. Продемонстрированы исследовательские умения и навыки. Фактических ошибок, связанных с пониманием проблемы, нет. Допущены одна-две ошибки в оформлении работы.

3 балла – студент проводит достаточно самостоятельный анализ основных этапов и смысловых составляющих проблемы; понимает базовые основы и теоретическое обоснование выбранной темы. Проблема раскрыта не полностью. Выводы не сделаны и/или выводы не обоснованы Проведен анализ проблемы без привлечения дополнительной литературы. Допущено не более 2 ошибок в смысле или содержании проблемы, оформлении работы.

2 балла - работа представляет собой пересказанный или полностью переписанный исходный текст без каких бы то ни было комментариев, анализа. Не раскрыта структура и теоретическая составляющая темы. Проблема не раскрыта. Выводы отсутствуют. Допущено три или более трех ошибок в смысловом содержании раскрываемой проблемы, в оформлении работы.

Критерии оценивания тестового задания

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Количество правильных ответов	91 % и ≥	от 81% до 90,9 %	не менее 70%	менее 70%

Критерии выставления оценки студенту на зачете/ экзамене

(оценочные средства: *устный опрос в форме ответов на вопросы билетов, устный опрос в форме собеседования, выполнение письменных разноуровневых задач и заданий, комплексная расчетно-графическая работа, творческое задание, кейс-задача, портфолио, проект и т.п.)*

Оценка по промежуточной аттестации	Характеристика качества сформированности компетенций
«зачтено» / «отлично»	Студент демонстрирует сформированность дисциплинарных компетенций на продвинутом уровне: обнаруживает всестороннее, систематическое и глубокое знание учебного материала, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
«зачтено» / «хорошо»	Студент демонстрирует сформированность дисциплинарных компетенций на базовом уровне: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
«зачтено» / «удовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на пороговом уровне: имеет знания только основного материала, но не усвоил его деталей, в ходе контрольных мероприятий допускаются значительные ошибки, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ, при оперировании знаниями и умениями при их переносе на новые ситуации.
«не зачтено» / «неудовлетворительно»	Студент демонстрирует сформированность дисциплинарных компетенций на уровне ниже порогового: выявляется полное или практически полное отсутствие знаний значительной части программного материала, студент допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, умения и навыки не сформированы.

5. Примеры оценочных средств для проведения текущей аттестации

5.1 Примеры тем для практических работ

1. Разработка простого программного модуля на основе линейного алгоритма.
2. Создание программного модуля с использованием условных операторов.
3. Реализация циклических алгоритмов в программном модуле.
4. Разработка модулей с использованием функций и процедур.
5. Создание классов и объектов в объектно-ориентированном программировании.
6. Разработка программного модуля обработки пользовательского ввода.
7. Реализация простого игрового механизма (движение объекта, столкновения).
8. Разработка модуля управления персонажем в игровой сцене.
9. Интеграция нескольких программных модулей в единое приложение.
10. Использование системы контроля версий Git при разработке программного модуля.
11. Отладка и исправление ошибок в программном коде.
12. Разработка и проведение модульного тестирования программного продукта.
13. Оптимизация программного модуля по производительности.
14. Создание интерфейса взаимодействия между программными модулями.

5.2 Примеры тестовых заданий

1. Что такое программный модуль?
   A) Графический интерфейс программы
   B) Самостоятельная часть программы, выполняющая определённую функцию
   C) Операционная система
   D) База данных
   Правильный ответ: B
2. Какой принцип НЕ относится к ООП?
   A) Инкапсуляция
   B) Наследование
   C) Компиляция
   D) Полиморфизм
   Правильный ответ: C
3. Какой инструмент используется для контроля версий кода?
   A) Photoshop
   B) Git
   C) Blender
   D) Excel
   Правильный ответ: B
4. Что такое отладка программы?
   A) Написание кода
   B) Удаление программы
   C) Поиск и исправление ошибок
   D) Установка программы
   Правильный ответ: C
   2. Выбор нескольких правильных ответов
5. Какие из перечисленных относятся к принципам ООП?

A) Инкапсуляция
B) Наследование
C) Компиляция
D) Полиморфизм
Правильные ответы: A, B, D

6. Какие инструменты используются при разработке программных модулей?
A) IDE
B) Git
C) Текстовый редактор
D) Игровой движок
Правильные ответы: A, B, D

3. Установление соответствия

7. Установите соответствие:

Понятие	**Определение**
1. Класс	A. Экземпляр класса
2. Объект	B. Шаблон для создания объектов
3. Метод	C. Функция внутри класса

Ответ: 1–B, 2–A, 3–C

4. Верно / неверно

8. Git используется для хранения графики в играх.
Ответ: Неверно
9. Инкапсуляция скрывает внутренние данные объекта.
Ответ: Верно
10. Программный модуль всегда зависит от других модулей.
Ответ: Неверно

5. Краткий ответ

11. Дайте определение программного модуля.
Ответ: самостоятельная часть программы, выполняющая определённую задачу.
12. Перечислите этапы жизненного цикла программного модуля.
Ответ: анализ, проектирование, реализация, тестирование, сопровождение.

6. Примеры оценочных средств для проведения промежуточной аттестации

6.1 Варианты вопросов для диффер. зачета

1. Что такое программный модуль?
2. Какие преимущества даёт модульная структура программы?
3. Назовите основные этапы жизненного цикла программного модуля.
4. Что включает этап анализа при разработке ПО?
5. Что такое проектирование программного модуля?
6. Какие инструменты используются для разработки программного обеспечения?
7. Для чего используется IDE?
8. Что такое система контроля версий?

9. Какие возможности предоставляет Git?
10. Что такое объектно-ориентированное программирование?
11. Перечислите основные принципы ООП.
12. Что такое класс в программировании?
13. Что такое объект?
14. Что такое метод класса?
15. Что такое инкапсуляция?
16. Что такое наследование?
17. Что такое полиморфизм?
18. В чём отличие структурного и объектно-ориентированного подхода?
19. Что такое алгоритм?
20. Какие виды алгоритмов вы знаете?
21. Что такое тестирование программного модуля?
22. Какие виды тестирования существуют?
23. Что такое отладка программы?
24. Какие инструменты используются для отладки?
25. Что такое логирование?
26. Для чего нужно сопровождение программного продукта?
27. Что такое оптимизация кода?
28. Какие способы оптимизации вы знаете?
29. Что такое рефакторинг кода?
30. Какие ошибки встречаются в программных модулях?
31. Что такое игровая механика?
32. Приведите примеры игровых механик.
33. Что такое игровой движок?
34. Какие игровые движки вы знаете?
35. Что такое сцена в игровом движке?
36. Как осуществляется интеграция программных модулей?
37. Что такое API?
38. Для чего используется взаимодействие модулей?
39. Что такое производительность программы?
40. Какие методы повышения производительности вы знаете?

6.2 Примеры тем с пояснениями для лабораторных работ

1. Разработка простого программного модуля на языке программирования. Выполняется создание базового модуля с реализацией линейного алгоритма и проверкой его работоспособности.

2. Реализация условных операторов в программном модуле. Разрабатывается модуль с использованием конструкций if/else для обработки различных сценариев.

3. Использование циклических конструкций в программировании. Создание модуля, выполняющего повторяющиеся действия с применением циклов for, while.

4. Разработка функций и процедур в программном модуле. Реализуется декомпозиция задачи на функции для повышения читаемости и повторного использования кода.

5. Создание классов и объектов (ООП). Разработка программного модуля с использованием классов, полей и методов.

6. Реализация принципов инкапсуляции. Создание класса с закрытыми полями и методами доступа (getter/setter).

7. Разработка простого игрового механизма. Реализуется движение объекта, обработка ввода и взаимодействие с простыми элементами сцены.

8. Обработка событий и пользовательского ввода. Создание модуля, реагирующего на действия пользователя (клавиатура, мышь).

9. Интеграция программных модулей в единую систему. Объединение нескольких модулей в одно приложение с проверкой взаимодействия.

10. Использование системы контроля версий Git при разработке проекта. Выполнение базовых операций (commit, branch, merge) в процессе разработки.

11. Отладка и исправление ошибок программного кода. Поиск логических и синтаксических ошибок с использованием инструментов отладки.

12. Оптимизация программного модуля. Улучшение производительности кода за счёт рефакторинга и устранения избыточных операций.