Министерство образования Самарской области

АССОЦИАЦИЯ ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ "ИНСТИТУТ НАПРАВО.РУ"

УТВЕРЖДЕНО
Приказ директора
от 07.08.2024 № 012/2/с

Воробьев К.А.

РАБОЧАЯ ПРОГРАММА ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

ПМ.01 Разработка программных модулей

Специальность 09.02.10
Разработка компьютерных игр, дополненной и виртуальной реальности

Квалификация: разработчик компьютерных игр, дополненной и виртуальной реальности
Очная форма обучения
На базе основного общего образования
Срок получения образования: 3 года 10 месяцев

Тольятти, 2024г.

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

ПМ.01 Разработка программных модулей

1.1. Цель и место профессионального модуля в структуре образовательной программы

Цель модуля: овладение видом профессиональной деятельности «Разработка программных модулей». Данный модуль направлен на формирование у студентов базовых компетенций в области разработки программного обеспечения, включающих алгоритмизацию задач, написание и оформление программного кода, отладку, оптимизацию и интеграцию программных модулей.

Профессиональный модуль включен в обязательную часть образовательной программы.

1.2. Планируемые результаты освоения профессионального модуля

Результаты освоения профессионального модуля соотносятся с планируемыми результатами освоения образовательной программы, представленными в матрице компетенций выпускника.

В результате освоения профессионального модуля обучающийся должен:

Код ОК, ПК	Уметь	Знать	Владеть навыками
ОК 01	распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части определять этапы решения задачи, составлять план действия, реализовывать составленный план, определять необходимые ресурсы выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы владеть актуальными методами работы в профессиональной и смежных сферах оценивать результат и последствия своих	актуальный профессиональный и социальный контекст, в котором приходится работать и жить структура плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях основные источники информации и ресурсы для решения задач и/или проблем в профессиональном и/или социальном контексте методы работы в профессиональной и смежных сферах порядок оценки результатов решения задач профессиональной деятельности	-
	действий (самостоятельно или с помощью наставника)		

Код ОК, ПК	Уметь	Знать	Владеть навыками
ОК 02	определять задачи для поиска информации, планировать процесс поиска, выбирать необходимые источники информации выделять наиболее значимое в перечне	номенклатура информационных источников, применяемых в профессиональной деятельности приемы структурирования	-

Код ОК, ПК	Уметь	Знать	Владеть навыками
	информации, структурировать получаемую информацию, оформлять результаты поиска оценивать практическую значимость результатов поиска применять средства информационных технологий для решения профессиональных задач использовать современное программное обеспечение в профессиональной деятельности использовать различные цифровые средства для решения профессиональных задач	информации формат оформления результатов поиска информации современные средства и устройства информатизации, порядок их применения и программное обеспечение в профессиональной деятельности, в том числе цифровые средства	
ОК 03	определять актуальность нормативно-правовой документации в профессиональной деятельности применять современную научную профессиональную терминологию определять и выстраивать траектории профессионального развития и самообразования	содержание актуальной нормативно-правовой документации современная научная и профессиональная терминология возможные траектории профессионального развития и самообразования основы предпринимательской деятельности, правовой и финансовой грамотности	-
	выявлять достоинства и недостатки коммерческой идеи определять инвестиционную привлекательность коммерческих идей в рамках профессиональной деятельности, выявлять источники финансирования презентовать идеи открытия собственного дела в профессиональной деятельности определять источники достоверной правовой информации составлять различные правовые документы находить интересные проектные идеи, грамотно их формулировать и документировать оценивать жизнеспособность проектной идеи, составлять план проекта	правила разработки презентации основные этапы разработки и реализации проекта	

Код ОК, ПК	Уметь	Знать	Владеть навыками
ОК 04	организовывать работу коллектива и команды взаимодействовать с коллегами, руководством, клиентами в ходе профессиональной деятельности	психологические основы деятельности коллектива психологические особенности личности	-
ОК 05	грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке проявлять толерантность в рабочем коллективе	правила оформления документов правила построения устных сообщений особенности социального и культурного контекста	-
ОК 06	проявлять гражданско-патриотическую позицию	сущность гражданско-патриотической позиции традиционных общечеловеческих	-
	демонстрировать осознанное поведение описывать значимость своей специальности применять стандарты антикоррупционного поведения	ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений значимость профессиональной деятельности по специальности стандарты антикоррупционного поведения и последствия его нарушения	
ОК 07	соблюдать нормы экологической безопасности определять направления ресурсосбережения в рамках профессиональной деятельности по специальности организовывать профессиональную деятельность с соблюдением принципов бережливого производства организовывать профессиональную деятельность с учетом знаний об изменении климатических условий региона эффективно действовать в чрезвычайных ситуациях	правила экологической безопасности при ведении профессиональной деятельности основные ресурсы, задействованные в профессиональной деятельности пути обеспечения ресурсосбережения принципы бережливого производства основные направления изменения климатических условий региона правила поведения в чрезвычайных ситуациях	-
Код ОК, ПК	**Уметь**	**Знать**	**Владеть навыками**
ОК 08	использовать физкультурно-оздоровительную деятельность для укрепления	роль физической культуры в общекультурном, профессиональном и	-

Код ОК, ПК	Уметь	Знать	Владеть навыками
	здоровья, достижения жизненных и профессиональных целей применять рациональные приемы двигательных функций в профессиональной деятельности	социальном развитии человека основы здорового образа жизни условия профессиональной деятельности и зоны риска физического здоровья для специальности	
	пользоваться средствами профилактики перенапряжения, характерными для данной специальности	средства профилактики перенапряжения	
ОК 09	понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы участвовать в диалогах на знакомые общие и профессиональные темы строить простые высказывания о себе и о своей профессиональной деятельности кратко обосновывать и объяснять свои действия (текущие и планируемые) писать простые связные сообщения на знакомые или интересующие профессиональные темы	правила построения простых и сложных предложений на профессиональные темы основные общеупотребительные глаголы (бытовая и профессиональная лексика) лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности особенности произношения правила чтения текстов профессиональной направленности	-
ПК 1.1	- Использовать методы и приемы формализации поставленных задач; - Использовать методы и приемы алгоритмизации поставленных задач; - Использовать программное обеспечение для графического отображения алгоритмов; - Применять алгоритмы решения типовых задач в соответствующих областях; - Осуществлять коммуникации с заинтересованными сторонами	- Методы и приемы формализации поставленных задач; - Языки формализации функциональных спецификаций; - Методы и приемы алгоритмизации поставленных задач; - Нотацию и программное обеспечение для графического отображения алгоритмов; - Алгоритмы решения типичных задач, области и способы их применения	- Составления формализованных описаний решений, поставленных задач в соответствии с требованиями технического задания или внутренних документов организации; - Разработки алгоритмов решения, поставленных задач в соответствии с требованиями технического задания или внутренних документов организации
ПК 1.2	- Применять выбранные языки программирования для	- Синтаксис выбранного языка программирования	- Разработки кода программного продукта на основе готовой

Код ОК, ПК	Уметь	Знать	Владеть навыками
	написания программного кода; - Использовать выбранную среду программирования и средства системы управления базами данных; - Использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода; - Осуществлять разработку кода программного модуля на языках низкого и высокого уровней; - Создавать программу по разработанному алгоритму как отдельный модуль	особенности программирования на этом языке, стандартные библиотеки языка программирования; - Методологию разработки компьютерного программного обеспечения; - Методологии и технологии проектирования и использования баз данных; - Технологии программирования; - Основные принципы технологии структурного и объектно-ориентированного программирования; - Особенности выбранной среды программирования и системы управления базами данных; - Компонент программно- технических архитектур, существующих приложений и интерфейсов взаимодействия с ними	спецификации на уровне модуля; - Использования инструментальных средств на этапе отладки программного продукта
ПК 1.3	- Применять нормативно-технические документы (стандарты и регламенты), определяющие требования к оформлению программного кода; - Применять инструментарий для создания и актуализации исходных текстов программ; - Применять заданные стандарты и шаблоны для составления и	- Инструментарий для создания и актуализации исходных текстов программ; - Методы повышения читаемости программного кода; - Системы кодировки символов, форматы хранения исходных текстов программ; - Нормативно-технические документы (стандарты и регламенты), определяющие требования к	- Приведения наименований переменных, функций, классов, структур данных и файлов в соответствие с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода; - Структурирования исходного
Код ОК, ПК	**Уметь**	**Знать**	**Владеть навыками**
	оформления технической документации	оформлению программного кода	программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода; - Комментирования и разметки программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного

Код ОК, ПК	Уметь	Знать	Владеть навыками
			кода; - Форматирования исходного программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода
ПК 1.4	- Использовать выбранную систему управления версиями; - Использовать вспомогательные инструментальные программные средства для обработки исходного текста программного кода	- Возможности используемой системы управления версиями и вспомогательных инструментальных программных средств; - Установленный регламент использования системы управления версиями	- Регистрации изменений исходного текста программного кода в системе управления версиями; - Слияния, разделения и сравнения исходных текстов программного кода; - Сохранения сделанных изменений программного кода в соответствии с регламентом управления версиями
ПК 1.5	- Выявлять ошибки в программном коде; - Применять методы и приемы отладки программного кода; - Интерпретировать сообщения об ошибках, предупреждения, записи технологических журналов; - Применять современные компиляторы, отладчики и оптимизаторы программного кода	- Методы и приемы отладки программного кода; - Типы и форматы сообщений об ошибках, предупреждений; - Способы использования технологических журналов, форматы и типы записей журналов	- Анализа и проверки исходного программного кода; - Отладки программного кода на уровне программных модулей; - Отладки программного кода на уровне межмодульных взаимодействий и взаимодействий с окружением
Код ОК, ПК	**Уметь**	**Знать**	**Владеть навыками**
ПК 1.6	- Применять методы, средства рефакторинга, оптимизации и инспекции программного кода; - Применять инструментальные средства коллективной работы над программным кодом; - Публиковать результаты рефакторинга, оптимизации и инспекции в коллективной базе знаний; - Интерпретировать сообщения, предупреждения, записи технологических журналов об ошибках, возникающих при выполнении дефектного кода	- Методы и средства рефакторинга, оптимизации и инспекции программного кода; - Методы и приемы отладки программного кода; - Типичные ошибки, возникающие при разработке компьютерного программного обеспечения, методы их диагностики и исправления; - Языки программирования и среды разработки; - Нормативно-технические документы (стандарты и регламенты), регламентирующие требования к программному коду, порядок отражения	- Анализа программного кода на соответствие требованиям по читаемости и производительности; - Инспекции программного кода для поиска не обнаруженных на ранних стадиях разработки компьютерного программного обеспечения ошибок и критических мест; - Внесения изменений в программный код и проверка его работоспособности; - Воспроизведения дефектов программного кода, зафиксированных в базе данных дефектов; - Установления причин

Код ОК, ПК	Уметь	Знать	Владеть навыками
		изменений в системе управления версиями, порядок отражения результатов рефакторинга, оптимизацию и инспекцию в коллективной базе знаний	возникновения дефектов программного кода
ПК 1.7	- Выбирать платформу/фреймворк в соответствии с требованиями к итоговому продукту; - Писать программный код процедур интеграции программных модулей; - Использовать выбранную среду программирования для разработки процедур интеграции программных модулей; - Применять методы и средства сборки модулей и компонентов компьютерного программного обеспечения, разработки процедур для развертывания компьютерного программного обеспечения, миграции и преобразования данных, создания программных интерфейсов	- Принципы и технологии функционирования современных платформ/фреймворков; - Методы и средства сборки модулей и компонентов компьютерного программного обеспечения; - Интерфейсы взаимодействия с внешней средой; - Интерфейсы взаимодействия внутренних модулей системы; - Методы и средства миграции и преобразования данных; - Языки, утилиты и среды программирования, средства пакетного выполнения процедур	- Анализа и выявления проблем сопряжения неоднородных модулей и компонентов компьютерного программного обеспечения; - Разработки и документирования программных интерфейсов; - Разработки процедур сборки модулей и компонентов компьютерного программного обеспечения; - Разработка процедур миграции и преобразования (конвертации) данных
ПК 1.8	- Выполнять процедуры сборки программных модулей и компонентов в программный продукт; - Производить настройки параметров платформы/фреймворка и осуществлять запуск процедур сборки; - Выполнять отладку программных модулей с использованием специальных средств платформ/фреймворков	- Методы и средства сборки и интеграции программных модулей и компонентов; - Интерфейсы взаимодействия программного продукта с внешней средой; - Интерфейсы взаимодействия внутренних модулей программного продукта; - Языки, утилиты и среды программирования, средства пакетного выполнения процедур	- Сборки программных модулей и компонентов в программный продукт; - Подключения программного продукта к компонентам внешней среды; - Внесения изменений в процедуры сборки модулей и компонентов компьютерного программного обеспечения, развертывания компьютерного программного обеспечения, миграции и преобразования данных; - Использования специализированных инструментов

платформы/фреймворка для отладки кода ПК 1.9

- Инсталлировать и настраивать аппаратное обеспечение, необходимое для функционирования платформ/фреймворков

- Производить развертывание и настройку платформ/фреймворков

- Характеристики и требования платформ/фреймворков к аппаратному обеспечению

- Разработки процедур развертывания и обновления компьютерного программного обеспечения

- Методы и средства разработки процедур для развертывания компьютерного программного обеспечения

- Основы системного администрирования

- Основы современных операционных систем

- Развертывания и настройки платформ/фреймворков

2. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

2.1. Объём и виды учебной работы

Вид учебной работы	Академические часы
Общий объём	340
Теоретическое обучение	82
Практические и лабораторные занятия	146
Курсовое проектирование	
Консультации	4
Самостоятельная работа	
Учебная практика	36
Производственная практика	72
Обязательная часть	288
Вариативная часть	52

Формы промежуточной аттестации компонентов модуля: Экзамен — семестр 5; Дифференцированный зачёт — семестр 5.

2.2. Структура профессионального модуля

Индекс и наименование	Всего	Теория	Практические занятия	Курсовой проект	Консультации	Практика
МДК.01.01 Разработка алгоритмов и программного кода	96	36	58		2	
МДК.01.02 Оптимизация программного кода	96	36	58		2	
МДК.01.03 Интеграция программных модулей и платформ/фреймворков	40	10	30			
УП.01.01 Учебная практика	36					36
ПП.01.01 Производственная практика	72					72
Итого	340	82	146		4	108

2.3. Содержание профессионального модуля

Раздел, тема	Содержание учебного материала и практических занятий	Часы
Раздел 1. Разработка алгоритмов и программного кода		96
МДК 01.01. Разработка алгоритмов и программного кода		96
Тема 1.1. Формализация задач и алгоритмизация	Содержание	12
	Понятие задачи и требования технического задания. Методы формализации поставленных задач (описание задачи на естественном языке, выделение исходных данных и результатов). Алгоритм как последовательность шагов решения задачи; свойства алгоритмов. Способы описания алгоритмов: словесный, псевдокод, блок-схемы. Базовые структуры алгоритмов: следование, ветвление, цикл. Типовые алгоритмы обработки данных (поиск, сортировка и др.). Использование программ для графического представления алгоритмов (редакторы блок-схем).	4
	В том числе практических занятий	8
	1. Разработка алгоритма для поставленной задачи по техническому заданию	2
	2. Анализ нескольких алгоритмов решения одной задачи и выбор оптимального по числу шагов	2
	3. Оформление описания алгоритма в виде схемы и текстового описания, удовлетворяющего требованиям внутреннего стандарта организации	4
Тема 1.2. Типы данных и структуры данных	Содержание	10
	Представление данных в памяти компьютера. Базовые типы данных (числовые, логические, символы и строки) и их диапазоны. Составные типы: массивы, записи (структуры), списки. Динамические структуры данных: списки, очереди, стеки, деревья (общее понятие). Алгоритмы обработки структур данных (добавление, поиск, удаление элементов). Понятие абстрактных типов данных.	4
	Оценка эффективности алгоритмов и операций над структурами (время и память на основные операции).	
	В том числе практических занятий	6
	4. Описание структуры данных для заданной предметной области	2
	5. Реализация простых операций над массивом (поиск максимума, сортировка)	2
	6. Анализ времени выполнения операций сортировки массива различными алгоритмами на небольшом наборе данных	2

Раздел, тема	Содержание учебного материала и практических занятий	Часы
Тема 1.3. Языки программирования и среды разработки	Содержание	12
	Классификация языков программирования: низкого уровня (assembler), языки среднего и высокого уровня. Обзор современного языка программирования, используемого в обучении: синтаксис, основные конструкции (операторы, условия, циклы, функции), понятие компиляции и интерпретации. Стандартные библиотеки языка и их назначение. Интегрированная среда разработки (IDE): возможности IDE для написания, отладки и сборки кода. Работа с системой управления базами данных (СУБД) из среды разработки. Компоненты программно-технической архитектуры приложений: исполняемый файл, библиотеки, модули, интерфейсы взаимодействия между ними.	4
	В том числе практических занятий	8
	7. Настройка рабочей среды: установка выбранной IDE и необходимых SDK/библиотек	2
	8. Написание и запуск простой программы («Hello, World!») на выбранном языке	2
	9. Использование возможностей IDE (подсветка синтаксиса, отладчик) для обнаружения и исправления синтаксических ошибок в коде	2
	10. Подключение к простой базе данных (например, SQLite) из кода: выполнение в программе базовых операций (создание таблицы, добавление записи, чтение данных)	2
Тема 1.4. Базы данных и хранение информации	Содержание	12
	Роль систем управления базами данных (СУБД) в программных приложениях. Модели данных: реляционная модель (таблицы, записи, поля, ключи). Язык SQL: основы составления запросов (SELECT, INSERT, UPDATE, DELETE). Проектирование простой базы данных по техническому заданию: определение сущностей и связей. Интеграция базы данных с приложением: технологии доступа к данным (например, ORM, API СУБД).	4
	Основы использования СУБД в приложениях: подключение, выполнение запросов, обработка результатов.	
	В том числе практических занятий	8
	11. Проектирование небольшой базы данных под заданную задачу (например, хранение настроек игры или результатов пользователей)	2
	12. Реализация функций доступа к данным из программы: запись данных в БД и считывание	4
	13. Выполнение SQL-запросов к разработанной базе данных и анализ полученных результатов	4
Тема 1.5. Принципы структурного и объектно-ориентированного программирования	Содержание	14

Раздел, тема	Содержание учебного материала и практических занятий	Часы
	Понятие структурного программирования: модульность, топ-даун проектирование, использование функций/процедур. Основные принципы объектно-ориентированного программирования (ООП): инкапсуляция, наследование, полиморфизм. Классы и объекты: свойства, методы, конструкторы. Организация программного кода в виде классов и модулей. Сравнение структурного и ООП-подходов, области применения каждого. Простые примеры классов (например, класс игрового персонажа с полями и методами). Основы проектирования классов и взаимодействия объектов. UML-диаграммы классов и последовательности — на простом уровне для понимания.	6
	В том числе практических занятий	8
	14. Разработка класса по описанию (например, класс «Персонаж игры» с заданными полями и методами) и тестирование его методов	2
	15. Реализация наследования: создание подкласса от базового класса и демонстрация переопределения методов	2
	16. Преобразование процедурного кода в объектно-ориентированный (группировка связанных функций и данных в класс)	2
	17. Составление простой UML-диаграммы классов для описания отношений между несколькими объектами в программе	2
Тема 1.6. Оформление программного кода и документация	Содержание	12
	Требования к качеству исходного кода. Нормативно-технические документы, регулирующие оформление программ (стандарты кодирования, корпоративные соглашения).	6
	Правила именования идентификаторов: понятные имена для переменных, функций, классов в соответствии со стандартами. Структурирование кода: разбиение на функции/методы, файлы, модули; соблюдение отступов, форматирование блоков кода. Комментирование кода: виды комментариев, стиль написания комментариев, генерация документации из комментариев (например, XML-документирование в C#). Оформление технической документации к программному модулю: спецификация, описание API, инструкция по использованию. Инструменты автоматического форматирования кода и отслеживания соответствия стилю (линтеры, встроенные средства IDE).	
	В том числе практических занятий	6
	18. Приведение исходного кода, написанного обучающимся (или предоставленного преподавателем), к соответствию заданному стилю кодирования: переименование переменных и функций по стандарту, перераспределение кода по функциям, добавление необходимых комментариев	2
	19. Использование инструмента форматирования кода для автоматического исправления стиля и анализ отчетов инструмента	2
	20. Составление простого технического описания к разработанному программному модулю (паспорт модуля: назначение, входные/выходные данные, автор, версия)	2

Раздел, тема	Содержание учебного материала и практических занятий	Часы
Тема 1.7. Системы контроля версий и коллективная разработка	Содержание	12
	Назначение системы контроля версий (СКВ) при разработке программных модулей. Обзор распространенных СКВ: Git, SVN (акцент на Git как индустриальном стандарте). Основные понятия Git: репозиторий, коммит, ветка, слияние, конфликт. Организация совместной работы с кодом: центральный и распределенный репозитории (например, платформа GitHub/GitLab). Стратегии ветвления: Git-flow, trunk-based development – обзор. Регламент ведения истории изменений кода в команде: частота коммитов, сообщения к коммитам, код-ревью. Инструменты для работы с Git (командная строка, графические клиенты). Решение конфликтов слияния и отката изменений.	4
	В том числе практических занятий	8
	21. Инициализация локального репозитория и сохранение версий кода: выполнение последовательности коммитов при разработке небольшого модуля	2
	22. Использование удаленного репозитория: клонирование проекта, создание ветки, выполнение изменений и отправка (push) на сервер	2
	23. Моделирование ситуации коллективной разработки: слияние (merge) изменений из двух веток, разрешение конфликта в коде	2
	24. Оформление changelog (списка изменений) по истории коммитов	2
Тема 1.8. Отладка программных модулей	Содержание	10
	Понятие отладки программного кода. Типы ошибок: синтаксические, логические, runtime-ошибки. Средства отладки в среде разработки: пошаговое выполнение кода, точки останова (breakpoints), просмотр значений переменных во время выполнения. Методы и приёмы отладки программного кода: двоичный поиск ошибки, вывод диагностических сообщений, логирование. Интерпретация сообщений об ошибках и предупреждений компилятора/исполняющей среды. Отладка на уровне модуля и взаимодействия модулей с окружением (файловой системой, СУБД, сетью). Инструменты отладки и профилирования (debugger, profiler) и их применение для нахождения узких мест.	4
	В том числе практических занятий	6
	25. Диагностика и исправление ошибок в программном модуле: использование отладчика для поиска и устранения ошибок	2
	26. Использование логирования: внедрение вывода в консоль или лог-файл, анализ содержания для выявления причины ошибки	2
	27. Отладка взаимодействия двух модулей: запуск, настройка окружения и выявление нарушений во взаимодействии компонентов	2
Консультация		2
Промежуточная аттестация – экзамен		

Раздел, тема	Содержание учебного материала и практических занятий	Часы
Раздел 2. Оптимизация программного кода		96
МДК 01.02. Оптимизация программного кода		96
	Содержание	26
Тема 2.1. Рефакторинг кода и улучшение архитектуры	Понятие рефакторинга: перепроектирование внутренней структуры кода без изменения его внешнего функционала. Цели рефакторинга: повышение читаемости, облегчение сопровождения, устранение дублирования. Примеры рефакторинга: декомпозиция длинной функции, объединение дублирующихся фрагментов, упрощение условий. Инструменты рефакторинга в IDE: автоматическое переименование, извлечение метода, перенос блоков кода. Метрики качества кода, понятие «кода с запахом». Роль рефакторинга в жизненном цикле ПО. Использование систем версий при рефакторинге, код-ревью изменений коллегами.	10
	В том числе практических занятий	16
	1. Анализ исходного кода на проблемы читаемости и дублирования	4
	2. Составление плана рефакторинга с перечнем улучшений	4
	3. Проведение рефакторинга с применением инструментов IDE и проверкой работоспособности	4
	4. Сравнение кода до и после рефакторинга, фиксация результатов	4
Тема 2.2. Оптимизация и повышение производительности	Содержание	24
	Основы оптимизации программного кода: критерии эффективности (время выполнения, использование памяти). Методы оптимизации алгоритмов: выбор эффективных структур данных, алгоритмов (например, хеш-таблица вместо списка). Оптимизация кода: сокращение избыточных операций, экономия памяти, учет алгоритмической сложности. Инструменты компилятора для оптимизации. Профилировщики: выявление узких мест в производительности. Баланс между читаемостью кода и его эффективностью. Примеры оптимизации в игровых приложениях (оптимизация циклов рендеринга). Аппаратные ограничения и подходы к оптимизации (многопоточность, SIMD — обзорно).	10
	В том числе практических занятий	14
	5. Измерение производительности фрагмента кода до и после оптимизации	4
	6. Оптимизация алгоритма по заданию с подтверждением ускорения	4
	7. Анализ потребления памяти и изменение структуры данных для её	6
Тема 2.3. Инспекция кода и статический анализ	Содержание	22

Раздел, тема	Содержание учебного материала и практических занятий	Часы
	Понятие инспекции кода (code review): ручная проверка кода на ошибки, уязвимости, несоответствия стандартам. Организация процесса инспекции в команде: правила, чек-листы, этапы. Статический анализ кода: автоматическая проверка без выполнения программы. Инструменты статического анализа (линтеры, анализаторы безопасности). Типичные дефекты: неиспользуемые переменные, NullReference, ошибки стиля, утечки памяти. Использование результатов анализа для рефакторинга и улучшения качества. Ведение базы знаний о дефектах: регистрация проблем в трекере, документация.	8
	В том числе практических занятий	14
	8. Парная инспекция кода по чек-листу (анализ кода однокурсника)	4
	9. Применение инструмента статического анализа к проекту, интерпретация отчёта	4
	10. Составление списка дефектов и рекомендаций по их устранению	6
Тема 2.4. Отслеживание и исправление дефектов	Содержание	22
	Жизненный цикл дефекта ПО: обнаружение, регистрация, воспроизведение, исправление, повторное тестирование. Инструменты отслеживания ошибок (Jira, Redmine): баг-репорт, приоритет, ответственный, статусы. Методы воспроизведения дефектов: анализ логов, отладка с заданными условиями, автоматизация. Root Cause Analysis — анализ причин дефекта. Разработка исправления (патча), проверка на отсутствие побочных эффектов. Использование системы контроля версий при исправлении: ветки hotfix, фиксация версии. Документирование исправлений: базы знаний, вики, шаблоны описания решений.	8
	В том числе практических занятий	14
	11. Воспроизведение дефекта по баг-репорту: запуск с заданными входными данными	2
	12. Локализация причины ошибки и её исправление	4
	13. Регистрация информации об ошибке и исправлении в баг-трекере	4
	14. Обсуждение в группе типовых ошибок и способов их предотвращения	4
Консультация		2
Промежуточная аттестация – экзамен		
Раздел 3. Интеграция программных модулей и платформ/фреймворков		40
МДК.01.03 Интеграция программных модулей и платформ/фреймворков		40

Раздел, тема	Содержание учебного материала и практических занятий	Часы
Тема 3.1. Проектирование интеграции модулей	Содержание	12
	Понятие интеграции программных модулей. Проблемы совместимости: форматы данных, интерфейсы, технологии. Методы анализа и выявления проблем сопряжения. Выбор платформы/фреймворка с учетом требований ТЗ. Программные интерфейсы (API): структура, описание, протоколы. Разработка и документирование API для модулей. Сценарии сборки компонентов: makefile, проекты IDE, CI/CD. Миграция и преобразование данных при интеграции.	2
	В том числе практических занятий	10
	1. Анализ совместимости модуля с внешней системой	2
	2. Проектирование интерфейса взаимодействия модуля (API)	2
	3. Разработка сценария сборки нескольких модулей	2
	4. Создание утилиты для преобразования данных между модулями	4
Тема 3.2. Сборка и интеграция программного продукта	Содержание	12
	Практическая реализация интеграции модулей. Сборка модулей и компонентов в программный продукт. Использование систем автоматизированной сборки (MSBuild, CMake и др.). Подключение внешних библиотек и платформ. Выполнение процедур развертывания: установка в целевую среду. Настройка параметров платформы/фреймворка перед запуском. Отладка на уровне интеграции. Контроль корректности интеграции: тестирование, логи, диагностика ошибок.	2
	В том числе практических занятий	10
	5. Сборка проекта из нескольких модулей с использованием системы сборки	2
	6. Интеграция разработанного модуля в проект на выбранной платформе	4
	7. Проведение тестового запуска интегрированного приложения и устранение	4
Тема 3.3. Развертывание и сопровождение программного обеспечения	Содержание	14
	Инсталляция программного обеспечения: создание установочного комплекта или скрипта.	4

Раздел, тема	Содержание учебного материала и практических занятий	Часы
	Настройка программного окружения: установка необходимых библиотек, фреймворков, драйверов. Требования платформ/фреймворков к аппаратному и программному обеспечению. Основы системного администрирования: настройка ОС, управление сервисами, мониторинг. Обновление программного обеспечения: новые версии, патчи, миграция данных.	

Раздел, тема	Содержание учебного материала и практических занятий	Часы
	Документирование и автоматизация процесса развертывания: скрипты, CI/CD. Обслуживание программного обеспечения: резервное копирование, анализ логов, устранение сбоев.	
	В том числе практических занятий	10
	8. Подготовка инсталляционного пакета с необходимыми компонентами	2
	9. Развертывание приложения на тестовой системе и настройка окружения	2
	10. Написание скрипта автоматического развертывания	2
	11. Имитирование обновления программного обеспечения	2
	12. Сбор и анализ логов после развертывания, устранение проблем	2
Промежуточная аттестация – дифференцированный зачет		2
Раздел, тема	**Содержание учебного материала и практических занятий**	**Часы**
Учебная практика Виды работ: 1. Реализация алгоритмов обработки данных: сортировка, фильтрация, поиск по массивам и спискам. 2. Разработка консольного приложения с пользовательским интерфейсом для решения типовой задачи (например, калькулятор, система учёта товаров). 3. Подключение и работа с базой данных: создание таблиц, выполнение SQL-запросов, вывод данных. 4. Ведение репозитория проекта в системе контроля версий (Git): создание коммитов, работа с ветками, описание изменений. 5. Оформление исходного кода в соответствии с корпоративными стандартами: именование, структурирование, комментарии. 6.		36

Раздел, тема	Содержание учебного материала и практических занятий	Часы
Проведение отладки разработанных модулей с использованием средств IDE и логирования. 7. Разработка и документирование API для программного модуля: описание функций, параметров, возможных ошибок. 8. Первичное тестирование функций: составление тест-кейсов, ручная проверка, фиксация дефектов.		
Производственная практика Виды работ: 1. Разработка программного продукта в составе учебного или внешнего проекта: определение требований, проектирование архитектуры. 2. Интеграция отдельных модулей в единое приложение: обеспечение взаимодействия, устранение конфликтов. 3. Реализация процесса сборки и развёртывания программного продукта: подготовка скриптов, настройка окружения.		7

4. Оптимизация кода по метрикам производительности: ускорение алгоритмов, сокращение объёма занимаемой

памяти.

5. Рефакторинг существующего кода с анализом его архитектуры и устранением дублирования.

6. Проведение статического анализа и исправление выявленных предупреждений.

7. Разработка и реализация плана обновления продукта: подготовка патча, тестирование стабильности после

изменений.

8. Подготовка итоговой технической документации: паспорт модуля, инструкция пользователя, отчёт об исправлениях. Промежуточная аттестация– экзамен Всего 340

3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

3.1. Материально-техническое обеспечение

Мастерская «Разработки программных модулей», оснащенная: посадочные места по количеству обучающихся (столы, стулья) рабочее место преподавателя шкаф или полки для хранения учебной и методической литературы доска меловая/маркерная/интерактивная ноутбук / компьютер с периферией (лицензионное программное обеспечение (по), образовательный контент и система защиты от вредоносной информации) - Персональные компьютеры, соответствующие современным требованиям (многоядерный процессор, не менее 8 ГБ ОЗУ, достаточный объем HDD/SSD) мультимедийный проектор или телевизор современная интегрированная среда разработки (IDE) с поддержкой выбранного языка программирования (например, Microsoft Visual Studio/Visual Studio Code для C#, C++; JetBrains Rider; средства для Blueprints) установленная система контроля версий (Git) и доступ к платформе для размещения репозиториев (локальный Git-сервер или интернет-сервисы типа GitHub) СУБД (например, Microsoft SQL Server Express, MySQL/MariaDB, SQLite) для выполнения практических заданий по работе с базами данных инструменты для профилирования и отладки (встроенные в IDE или отдельные утилиты). При необходимости – инструменты статического анализа кода (типа SonarLint, PVS-Studio – с учебной лицензией) для проведения лабораторных работ по инспекции кода. специализированные утилиты профилирования и отладки (например, Unity Profiler, GPU-Z, сетевые эмуляторы задержек)

Оснащенные базы практики (мастерские/зоны по видам работ).

3.2. Информационное обеспечение

1. Подбельский В. В. Программирование. Базовый курс C# : учебник для вузов. Москва : Юрайт, 2024. 369 с. ISBN 978-5-534-10616-9. https://urait.ru/bcode/536775.

2. Кувшинов Д. Р. Основы программирования : учебное пособие для вузов. Москва : Юрайт, 2024. 104 с. ISBN 978-5-534-07559-5. https://urait.ru/bcode/540199.

3. Chacon S., Straub B. Pro Git. Second edition. Apress, 2014. https://git-scm.com/book/en/v2.

4. Unity Technologies. Unity User Manual 2022.3 (LTS). https://docs.unity3d.com/2022.3/Documentation/Manual/UnityManual.html.

5. Godot Engine. Godot Docs, версия 4.2. https://docs.godotengine.org/en/4.2/.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Код ПК, ОК	Критерии оценки результата (показатели освоенности компетенций)	Формы контроля и методы оценки
ПК 1.1	Алгоритм соответствует требованиям ТЗ, корректно решает задачу, оптимален по структуре; представлен в понятной форме (псевдокод/блок-схема) и может быть реализован программно.	Форма контроля: зачет, контрольная работа. Методы оценки: интерпретация результатов выполнения лабораторных заданий, оценка решения ситуационных задач.
ПК 1.2	Программный модуль выполняет все функции согласно ТЗ; код компилируется без ошибок, проходит тестовые примеры; используются подходящие типы данных, структуры, алгоритмы.	Форма контроля: зачет, защита курсовой работы. Методы оценки: оценка результата практического задания, интерпретация выполнения лабораторных работ.
ПК 1.3	Соблюдены стандарты именования, отступов и форматирования; комментарии поясняют сложные фрагменты; прилагается техническая документация.	Форма контроля: зачет. Методы оценки: интерпретация оформленного программного кода, рецензирование документации.
ПК 1.4	Все изменения зафиксированы коммитами с понятными сообщениями; выполнено слияние без конфликтов; освоены команды Git.	Форма контроля: зачет. Методы оценки: оценка результата лабораторной работы, анализ истории версий, решение ситуационных задач.
ПК 1.5	Студент способен локализовать и объяснить ошибки, предложить исправление; использует точки останова, стек вызовов, анализ логов.	Форма контроля: экзамен, зачет. Методы оценки: оценка выполнения лабораторного задания с ошибками, интерпретация результатов отладки.
ПК 1.6	Улучшенный код более читаем или быстрее работает; все тесты	Форма контроля: защита курсового проекта, зачет.
	проходят; изменения зафиксированы и документированы.	Методы оценки: анализ до- и после- версии кода, оценка решения практической задачи на оптимизацию.
ПК 1.7	Представлен план интеграции; интерфейс (API) корректен; сборка проходит успешно; миграция данных протестирована.	Форма контроля: зачет. Методы оценки: оценка проектной документации и демонстрации работы процедур интеграции.
ПК 1.8	Продукт собран и запускается; конфигурация среды выполнена; ошибки интеграции найдены и устранены.	Форма контроля: зачет, экзамен. Методы оценки: оценка выполнения практического задания, интерпретация результата сборки проекта.
ПК 1.9	Установка проходит без ошибок; ПО работает; окружение настроено; обновления применяются без потерь данных.	Форма контроля: квалификационное испытание, зачет. Методы оценки: оценка результата инсталляции и настройки ПО, решение ситуационных задач.
ОК 01	Анализ и подбор способов решения задач профессиональной деятельности.	Экспертное наблюдение выполнения практических работ Дифференцированный зачет Экзамен
ОК 02	Использование современное программные средства и информационные технологии для выполнения задач профессиональной деятельности.	Экспертное наблюдение выполнения практических работ Дифференцированный зачет Экзамен

Код ПК, ОК	Критерии оценки результата (показатели освоенности компетенций)	Формы контроля и методы оценки
ОК 03	Применение современной научной профессиональной терминологии.	□ Устный опрос
ОК 04	Взаимодействие с коллегами, руководством, клиентами в ходе профессиональной деятельности.	□ Экспертное наблюдение выполнения практических работ □ Дифференцированный зачет □ Экзамен
ОК 05	Грамотное изложение своих мыслей по профессиональной тематике на государственном языке, проявление толерантность в рабочем коллективе.	□ Устный опрос, Экспертное наблюдение выполнения практических работ □ Дифференцированный зачет □ Экзамен
ОК 06	Понимание значимости профессиональной деятельности по специальности	□ Устный опрос.
ОК 07	Содействие сохранению окружающей среды, ресурсосбережению, применение знаний о принципах бережливого производства.	□ Экспертное наблюдение выполнения практических работ □ Дифференцированный зачет □ Экзамен.
ОК 08	Использование средств профилактики перенапряжения, характерные для данной специальности	□ Экспертное наблюдение выполнения практических работ □ Дифференцированный зачет □ Экзамен

□ Устный опрос.

ОК 09 Понимание общего смысла четко произнесенных высказываний на известные темы.