

**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОБРАЗОВАТЕЛЬНАЯ
ОРГАНИЗАЦИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
ЦЕНТРОСОЮЗА РОССИЙСКОЙ ФЕДЕРАЦИИ
«РОССИЙСКИЙ УНИВЕРСИТЕТ КООПЕРАЦИИ»
КАМЧАТСКИЙ ФИЛИАЛ**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

**МДК 01.01 ВЫПОЛНЕНИЕ РАБОТЫ ПО ПОДГОТОВКЕ И
ОБРАБОТКЕ ДАННЫХ РАЗЛИЧНЫХ ФОРМАТОВ**

**ПМ.01 ОФОРМЛЕНИЕ И КОМПОНОВКА ТЕХНИЧЕСКОЙ
ДОКУМЕНТАЦИИ**

Профессия 09.01.03 Оператор информационных систем и ресурсов

Форма обучения: очная

Объем дисциплины: 113 ак.ч.

Форма промежуточной аттестации: экзамен

Оценочные материалы по МДК 01.01 Выполнение работы по подготовке и обработке данных различных форматов:

обсуждены и рекомендованы к утверждению решением кафедры экономики, технологии и таможенного дела от 04 апреля 2026 г., протокол № 3

одобрены Ученым советом филиала от 17 апреля 2026 г., протокол № 5

СОДЕРЖАНИЕ

1. Паспорт фонда оценочных средств по дисциплине
2. Оценочные материалы по дисциплине:
 - 2.1. Оценочные материалы для проведения текущего контроля.
 - 2.2. Оценочные материалы для проведения промежуточной аттестации.

Обновление рабочей программы дисциплины (модуля)

Целью оценочных материалов по дисциплине (модулю) (далее –ОМ) является комплексная оценка результатов обучения по дисциплине (модулю) МДК 01.01 Выполнение работы по подготовке и обработке данных различных форматов и установление уровня сформированности компетенций обучающегося.

ОМ предназначены для проведения текущего контроля успеваемости и промежуточной аттестации.

Ом включают оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации.

Оценочные материалы разработаны в соответствии с рабочей программой МДК 01.01 Выполнение работы по подготовке и обработке данных различных форматов:

.

1. Паспорт фонда оценочных средств по дисциплине «МДК 01.01 Выполнение работы по подготовке и обработке данных различных форматов»

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;	знать: - актуальный профессиональный и социальный контекст, в котором приходится работать и жить; - структура плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях; - 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 действий (самостоятельно или с помощью наставника) уметь: - распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части;	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (П1)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	- 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 действий (самостоятельно или с помощью наставника)			
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;	знания: Основные принципы чистого кода (Clean Code). Стандарты и практики документирования программного обеспечения. Инструменты для автоматической проверки качества кода (например, PyLint, ESLint). умения: Оформлять код в соответствии с принятыми стандартами и требованиями.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П2)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Документировать разработанный программный код. Применять соглашения о наименованиях переменных, функций и классов (например, PEP8 для Python). иметь практический опыт: 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).			
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой	Умения: Реализовывать программные модули на основе требований технического задания. Писать чистый, понятный и поддерживаемый код. Использовать стандартные библиотеки и фреймворки для ускорения разработки. Знания:	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (ПЗ)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
грамотности в различных жизненных ситуациях;	Принципы модульного программирования. Языки программирования для разработки модулей (Python, C#, Java). Стандартные фреймворки и библиотеки для работы с ИИ (TensorFlow, PyTorch, Keras). иметь практический опыт: Разработки модульных ИИ-систем, соответствующих требованиям производительности и безопасности. Внедрения разработанных ИИ-модулей в комплексные программные системы. Оптимизации кода и работы с интерфейсами для взаимодействия между модулями.	Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем		
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде;	знать: - основные методы и подходы к построению алгоритмов (жадные алгоритмы, динамическое программирование, рекурсивные подходы); -принципы эффективной обработки данных;	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (П4)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	- языки программирования, применяемые для разработки алгоритмов (Python, C#, Java). - уметь: - анализировать технические задания и выявлять требования к алгоритмам; - применять методы алгоритмизации для решения задач программирования; - разрабатывать оптимальные алгоритмы для решения задач в области ИИ. - иметь практический опыт: - 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.	Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем		
ОК 05. Осуществлять устную и письменную коммуникацию на	Умения:	Тема 1.1. Введение в искусственный интеллект и его направления	<i>Тестовое задание</i>	<i>Тест (П5)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;	Реализовывать программные модули на основе требований технического задания. Писать чистый, понятный и поддерживаемый код. Использовать стандартные библиотеки и фреймворки для ускорения разработки. Знания: Принципы модульного программирования. Языки программирования для разработки модулей (Python, C#, Java). Стандартные фреймворки и библиотеки для работы с ИИ (TensorFlow, PyTorch, Keras). иметь практический опыт: Разработки модульных ИИ-систем, соответствующих требованиям производительности и безопасности. Внедрения разработанных ИИ-модулей в комплексные программные системы. Оптимизации кода и работы с интерфейсами для взаимодействия между модулями.	Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Практическое задание</i>	

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных российских духовно-нравственных ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения;	знания: Основные принципы чистого кода (Clean Code). Стандарты и практики документирования программного обеспечения. Инструменты для автоматической проверки качества кода (например, PyLint, ESLint). умения: Оформлять код в соответствии с принятыми стандартами и требованиями. Документировать разработанный программный код. Применять соглашения о наименованиях переменных, функций и классов (например, PEP8 для Python). иметь практический опыт: Оформления, документирования и структурирования кода для последующей поддержки. Использования инструментов статического анализа кода для выявления ошибок и улучшения качества.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (П6)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Работы с системами документирования кода (например, Doxygen, Sphinx).			
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях;	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт: Управления проектами с использованием Git для организации командной работы. Разрешения конфликтов при слиянии веток и использования pull request для рецензирования кода.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (П7)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Настройки процессов CI/CD для автоматического тестирования и развертывания кода.			
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности;	знания: Основные принципы чистого кода (Clean Code). Стандарты и практики документирования программного обеспечения. Инструменты для автоматической проверки качества кода (например, PyLint, ESLint). умения: Оформлять код в соответствии с принятыми стандартами и требованиями. Документировать разработанный программный код. Применять соглашения о наименованиях переменных, функций и классов (например, PEP8 для Python). иметь практический опыт: Оформления, документирования и структурирования кода для последующей поддержки.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (П8)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Использования инструментов статического анализа кода для выявления ошибок и улучшения качества. Работы с системами документирования кода (например, Doxygen, Sphinx).			
ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт: Управления проектами с использованием Git для организации командной работы.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Разрешения конфликтов при слиянии веток и использования pull request для рецензирования кода. Настройки процессов CI/CD для автоматического тестирования и развертывания кода.			
ПК 1.1. Выполнять ввод и обработку текстовых данных.	знать: - основные методы и подходы к построению алгоритмов (жадные алгоритмы, динамическое программирование, рекурсивные подходы); - принципы эффективной обработки данных; - языки программирования, применяемые для разработки алгоритмов (Python, C#, Java). уметь: - анализировать технические задания и выявлять требования к алгоритмам; - применять методы алгоритмизации для решения задач программирования; - разрабатывать оптимальные алгоритмы для решения задач в области ИИ.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П18, П19, П25)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	иметь практический опыт: - 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. Выполнять преобразование данных, связанных с изменениями структуры документов.	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода.	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание</i> <i>Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	иметь практический опыт: Управления проектами с использованием Git 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.			
ПК 1.3. Выполнять разметку и форматирование документов различных форматов.	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт:	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Управления проектами с использованием Git 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.			
ПК 1.4. Конвертировать аналоговые данные в цифровые.	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт:	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Управления проектами с использованием Git 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.			
ПК 1.5. Выполнять подготовку цифровых данных для дальнейшей обработки и архивирования.	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт:	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Управления проектами с использованием Git 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.			
ПК 1.6. Формировать запросы для получения информации в базах данных.	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт:	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Управления проектами с использованием Git 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.			
ПК 1.7. Выполнять операции с объектами базы данных.	Знания: Принципы работы распределенных систем контроля версий. Основные команды и операции в Git (commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки. Умения: Работать с системами контроля версий для управления проектами (Git, GitLab). Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода. иметь практический опыт:	Тема 1.1. Введение в искусственный интеллект и его направления Тема 1.2 Методы сбора и предобработки данных Тема 1.3. Основы алгоритмов машинного обучения Тема 1.4. Оценка качества моделей и улучшение алгоритмов Тема 1.5. Глубокое обучение и нейронные сети Тема 1.6 Проектирование ИИ-систем	<i>Тестовое задание Практическое задание</i>	<i>Тест (П9)</i>

Формируемые компетенции (код и наименование компетенции)	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	3	4	5	6
	Управления проектами с использованием Git 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. Оценочные материалы по дисциплине *МДК 01.01 Выполнение работы по подготовке и обработке данных различных форматов*

2.1 Оценочные материалы для проведения текущего контроля успеваемости

ЗАДАНИЯ ДЛЯ ПРАКТИЧЕСКИХ РАБОТ В том числе в форме практической подготовки

Задание 1.

1. Разработать техническое задание по варианту, указанного преподавателем.
2. Оформить отчет

Вопросы для закрепления теоретического материала к практическому занятию:

1. Назовите этапы разработки программного обеспечения.
2. Что включает в себя постановка задачи и предпроектные исследования?
3. Перечислите функциональные и эксплуатационные требования к программному продукту.
4. Перечислите правила разработки технического задания.

Задание 2

На основе технического задания из практической работы №1 выполнить анализ функциональных и эксплуатационных требований к программному продукту.

Определить основные технические решения (выбор языка программирования, структура программного продукта, состав функций ПП, режимы функционирования) и занести результаты в документ, называемый «Эскизным проектом» (см. приложение 4).

Используя Приложение 5, выполнить следующее (по заданию преподавателя):

Определить диаграммы потоков данных для решаемой задачи.

Определить диаграммы «сущность-связь», если программный продукт содержит базу данных. Определить функциональные диаграммы.

Определить диаграммы переходов состояний. Определить спецификации процессов. Добавить словарь терминов. Оформить результаты, используя MS Office или MS Visio в виде эскизного проекта. Сдать и защитить работу.

Вопросы для закрепления теоретического материала к практическому занятию:

1. Назовите этапы разработки программного обеспечения.
2. Что такое жизненный цикл программного обеспечения?
3. В чем заключается постановка задачи и предпроектные исследования?
4. Назовите функциональные и эксплуатационные требования к программному продукту.
5. Перечислите составляющие эскизного проекта.
6. Охарактеризуйте спецификации и модели.

Задание 3

Для предметной области (заданной преподавателем к практической работе №2) выполнить объектно-ориентированное проектирование программного продукта.

2. Оформить отчет.

Отчет по практической работе должен быть оформлен на основании требований и состоять из следующих структурных элементов:

Анализа предметной области

Определения функций предметной области

Схемы документопотока

Выделенных сущностей, атрибутов и установленных связей

Концептуальной модели

Описания выходных и входных данных

Ответить на контрольные вопросы

Вопросы для закрепления теоретического материала к практическому занятию:

1. В чем заключается сущность объектно-ориентированного подхода при разработке программного продукта?
2. На что направлен объектно-ориентированный анализ?
3. Перечислите основные достоинства объектно-ориентированной методологии по сравнению со структурными методами.
4. Перечислите принципы объектного подхода. Дайте им краткие характеристики
5. Назовите основные методики объектно-ориентированного анализа.
6. Какие понятия, необходимые для проектирования системы включает концептуальная модель?

Критерии оценки выполнения практических работ по МДК

Оценка «отлично» выставляется студенту, если задание выполнено правильно;

Оценка «хорошо» выставляется студенту, если задание выполнено правильно, но присутствуют небольшие недочеты;

Оценка «удовлетворительно» выставляется студенту, если задание выполнено не полностью и присутствуют недочеты;

Оценка «неудовлетворительно» выставляется студенту, если задание не выполнено

ТЕСТОВОЕ ЗАДАНИЕ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ АТТЕСТАЦИИ

1. Операционная система:

- a) система программ, которая обеспечивает совместную работу всех устройств компьютера по обработке информации
- b) система математических операций для решения отдельных задач
- c) система планового ремонта и технического обслуживания компьютерной техники

2. Программное обеспечение (ПО) – это:

- a) совокупность программ, позволяющих организовать решение задач на компьютере
- b) возможность обновления программ за счет бюджетных средств
- c) список имеющихся в кабинете программ, заверен администрацией школы

3. Загрузка операционной системы – это:

- a) запуск специальной программы, содержащей математические операции над числами
- b) загрузка комплекса программ, которые управляют работой компьютера и организуют диалог пользователя с компьютером
- c) вложение дискеты в дисковод

4. Система программирования – это:

- a) комплекс любимых программ программиста
- b) комплекс программ, облегчающий работу программиста
- c) комплекс программ, обучающих начальным шагам программиста

5. Прикладное программное обеспечение – это:

- a) справочное приложение к программам
- b) текстовый и графический редакторы, обучающие и тестирующие программы, игры
- c) набор игровых программ

6. Прикладное программное обеспечение:

- a) программы для обеспечения работы других программ
- b) программы для решения конкретных задач обработки информации
- c) программы, обеспечивающие качество работы печатающих устройств

7. Операционные системы:

- a) DOS, Windows, Unix
- b) Word, Excel, Power Point
- c) (состав отделения больницы): зав. отделением, 2 хирурга, 4 мед. Сестры

8. Системное программное обеспечение:

- a) программы для организации совместной работы устройств компьютера как единой системы
- b) программы для организации удобной системы размещения программ на диске
- c) набор программ для работы устройства системного блока компьютера

9. Сервисные (обслуживающие) программы:

- a) программы сервисных организаций по бухгалтерскому учету
- b) программы обслуживающих организаций по ведению делопроизводства
- c) системные оболочки, утилиты, драйвера устройств, антивирусные и сетевые программы

10. Системные оболочки – это:

- a) специальная кассета для удобного размещения дискет с операционной системой
- b) специальная программа, упрощающая диалог пользователь – компьютер, выполняет команды операционной системы
- c) система приемов и способов работы конкретной программы при загрузке

программ и завершении работы

11. Использование одного имени для задания общих для класса действий, что означает способность объектов выбирать внутренний метод, исходя из типа данных, определяет свойство ООП

- a) Полиморфизм
- b) Управление событиями
- c) Инкапсуляция
- d) Наследование

12. Понятие «инкапсуляция» относится к

- a) Технологии модульного программирования
- b) Технологии объектно – ориентированного программирования
- c) Технологии императивного программирования
- d) Технологии модульного программирования

13.Свойство ООП, которое может быть смоделировано с помощью таксономической классификационной схемы (иерархии) называется

- a) Инкапсуляция
- b) Управление событиями
- c) Полиморфизм
- d) Наследование

14.Понятие класса в ООП включает в себя

- a) Поля и методы класса
- b) Процедуры и функции обработки
- c) Поля и функции обработки
- d) Поля и процедуры обработки

15.Назначение конструктора объекта

- a) Только выделяет память под объект
- b) Выделяет память и задает начальное значение полям
- c) Задает начальное значение полям
- d) Выделяет память, задает начальное значение полям, выполняет любые проверки, заданные программистом

16.Как описывается конструктор объекта

- a) procedure create;
- b) constructor create;
- c) function create;
- d) function constructor;

17. Как описывается деструктор объекта

- a) procedure free;
- b) destructor free;
- c) free;
- d) function free;

18. Понятия объекта в ООП - это

- a) представитель класса
- b) конкретные данные, заданные в классе.
- c) компонент панели инструментов
- d) встроенный объект Delphi

19. Моделями типа «черный ящик» являются

- a) Модели мышления
- b) Модели, описывающие зависимость параметров состояния объекта от входных параметров
- c) Модели, описывающие входные и выходные параметры объекта без учета внутренней структуры объекта
- d) Модели «аварийного» ящика на самолетах

20. Моделями типа «белый ящик» являются
- a) Модели мышления
 - b) Модели, описывающие зависимость параметров состояния объекта от входных параметров
 - c) Модели, описывающие входные и выходные параметры объекта с учетом внутренней структуры объекта
 - d) Модели, описывающие выходные данные в программе
21. Программа «драйвер» служит для
- a) запуска программы на выполнение
 - b) имитации запуска программы на выполнение
 - c) проверки правильности работы программы
 - d) передачи параметров в процедуры и функции
22. Программа «заглушка» служит для
- a) запуска программы на выполнение
 - b) имитации запуска другой программы на выполнение
 - c) проверки правильности работы программы
 - d) имитации передачи параметров в другой модуль
23. Какие методы сборки программы существуют
- a) монолитная
 - b) пошаговая
 - c) одновременная
 - d) постепенная
24. Какой метод тестирования программы учитывает закон распределения входных данных
- a) детерминированное тестирование
 - b) функциональное тестирование
 - c) стохастическое тестирование
 - d) логическое тестирование
25. Программирование сверху вниз – это
- a) Процесс, при котором от начального предположения осуществляется движение по направлению к лучшим решениям
 - b) Процесс пошагового разбиения алгоритма на все более мелкие части с целью получения таких элементов, для которых можно написать конкретные команды
 - c) Метод сведения трудной задачи к последовательности более простых
 - d) Исследование древовидной модели пространства решений и ориентация на поиск оптимального решения
26. Загрузочный модуль программы – результат работы
- a) Грамматики
 - b) Транслятора
 - c) Интерпретатора
 - d) Редактора связей (компоновщика)
27. Интегрированная система программирования включает компонент для перевода исходного текста программы в машинный код, который называется
- a) строителем кода
 - b) компилятор
 - c) переводчиком
 - d) преобразователем
28. Результатом компиляции программы на языке высокого уровня является
- a) Командный файл
 - b) Объектный файл
 - c) Исходный текст программы на языке высокого уровня

d) Дисплейный файл

Критерии оценки для проведения тестирования

Шкала оценивания результатов тестирования и критерии выставления оценок.

Оценка	Уровень сформированности компетенции	Критерии
«Отлично» («зачтено»)	Высокий	Выполнено более 80% заданий предложенного теста
«Хорошо» («зачтено»)	Хороший	Выполнено 60-80% заданий предложенного теста
«Удовлетворительно» («зачтено»)	Достаточный	Выполнено 35-60% заданий предложенного теста
«Неудовлетворительно» («не зачтено»)	Недостаточный	Выполнено менее 35% заданий предложенного теста

2.2 Оценочные материалы для проведения промежуточной аттестации

Вопросы к экзамену:

ОК.1 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам

1. Характеристика программы и программного обеспечения
2. Характеристика задач и приложений
3. Классификация программных продуктов
4. Объектно-ориентированные языки программирования, их характеристики
5. Проектирование пользовательского интерфейса

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности

1. Технологические и функциональные задачи
2. Процесс создания программ
3. Постановка задачи
4. Алгоритмизация
5. Характеристика программного продукта и его специфика

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста

1. Каскадная модель жизненного цикла
2. Поэтапная модель жизненного цикла с промежуточным контролем
3. Внутренняя организация программного обеспечения
4. Методологии разработки программного обеспечения
5. Определение «кодирование», его цель.

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием

1. Спиральная модель жизненного цикла
2. Основные процессы жизненного цикла
3. Модульное программирование
4. Структурное программирование
5. Стадии и этапы проектирования

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием

1. Вспомогательные процессы жизненного цикла
2. Организационные процессы жизненного цикла
3. Автоматизированное проектирование алгоритмов и программ
4. Разработка справочной системы программного обеспечения, основные характеристики
5. Понятие и сущность тестов

ПК 1.3 Оформлять программный код в соответствии с техническим заданием

1. Качество программного продукта
2. Классы прочности модуля
3. Виды тестов и сфера их применения
4. Неавтоматизированное проектирование алгоритмов и программ
5. Метод структурного проектирования: цель, принцип, преимущество

ПК 1.4 Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки

1. Критерии качества программных продуктов
2. Характеристики качества программного средства
3. Сопровождение программ
4. Принципы коллективной разработки программ
5. Методы коллективной разработки программ

ПК 1.5 Выполнять отладку программных модулей с использованием специализированных программных средств

1. Общие характеристики качества программных систем
2. Методы управления качеством, используемые в современных технологиях программирования
3. Организация коллективной разработки программистов

ПК 1.6. Выполнять тестирование программного кода

1. Аттестация программных систем
2. Функциональные требования к программной системе
3. Анализ требований
4. Что такое техническое задание? Для чего оно нужно?
5. Объектно-ориентированное программирование

ПК 1.7 Составлять тестовые сценарии

1. Нефункциональные требования к программной системе
2. Методы первичного сбора требований
3. Принципы системного проектирования
4. Нисходящее проектирование программ
5. Принципы модульного проектирования

Тестовые задания для проведения промежуточной аттестации в форме экзамена:

П1. Что из перечисленного относится к специфическим особенностям ПО как продукта:

- а. низкие затраты при дублировании;
- б. универсальность;
- в. простота эксплуатации;
- г. наличие поддержки (сопровождения) со стороны разработчика

П2. Этап, занимающий наибольшее время, в жизненном цикле программы:

- а. сопровождение;
- б. проектирование;
- в. тестирование;
- г. программирование;

П3. Этап, занимающий наибольшее время, при разработке программы:

- а. тестирование;
 - б. сопровождение;
 - в. проектирование;
 - г. программирование;
- П4. Первый этап в жизненном цикле программы:
- а. анализ требований;
 - б. формулирование требований;
 - в. проектирование;
 - г. автономное тестирование;
- П5. Один из необязательных этапов жизненного цикла программы:
- а. проектирование;
 - б. тестирование;
 - в. программирование;
 - г. оптимизация
- П6. Самый большой этап в жизненном цикле программы:
- а. эксплуатация;
 - б. изучение предметной области;
 - в. тестирование;
 - г. корректировка ошибок
- П7. Какой этап выполняется раньше:
- а. тестирование;
 - б. отладка;
 - в. эксплуатация;
 - г. оптимизация
- П8. Какой из этапов выполняется раньше остальных:
- а. отладка;
 - б. оптимизация;
 - в. программирование;
 - г. тестирование
- П9. Что выполняется раньше:
- а. компиляция;
 - б. отладка;
 - в. компоновка;
 - г. тестирование
- П10. В стадии разработки программы не входит:
- а. постановка задачи;
 - б. составление спецификаций;
 - в. автоматизация программирования;
 - г. эскизный проект
- П11. Самый важный критерий качества программы:
- а. надежность;
 - б. работоспособность;
 - в. быстродействие;
 - г. простота эксплуатации
- П12. Один из способов оценки качества ПО:

- а. сравнение с аналогами;
 - б. наличие документации;
 - в. оптимизация программы;
 - г. структурирование алгоритма
- П13. Существует ли связь между эффективностью и оптимизацией программы:
- а. да;
 - б. нет;
 - в. в случаях коллективной разработки ПО;
 - г. в случаях индивидуальной разработки ПО
- П14. Наиболее важным критерием качества при разработке ПО является:
- а. быстродействие;
 - б. удобство в эксплуатации;
 - в. надежность;
 - г. эффективность
- П15. Одним из способов оценки надежности ПО является:
- а. сравнение с аналогами;
 - б. трассировка;
 - в. оптимизация;
 - г. тестирование
- П16. В каких единицах можно измерить надежность разработанного программного продукта:
- а. отказов/час;
 - б. км/час;
 - в. Кбайт/сек;
 - г. операций/сек
- П17. В каких единицах можно измерить быстродействие ПО:
- а. отказов/час;
 - б. км/час;
 - в. Кбайт/сек;
 - г. операций/сек
- П18. Что относится к этапу программирования:
- а. написание кода программы;
 - б. разработка интерфейса;
 - в. работоспособность;
 - г. анализ требований.
- П19. Укажите правильную последовательность этапов программирования:
- а. компилирование, компоновка, отладка;
 - б. компоновка, отладка, компилирование;
 - в. отладка, компилирование, компоновка;
 - г. компилирование, отладка, компоновка.
- П20. К инструментальным средствам программирования относятся:
- а. компиляторы, интерпретаторы;
 - б. СУБД (системы управления базами данных);
 - в. BIOS (базовая система ввода-вывода);

- г. ОС (операционные системы).
- П21. Доступ, при котором записи файла читаются в физической последовательности, называется:
 - а. прямым;
 - б. простым;
 - в. последовательным;
 - г. основным
- П22. Какого метода программирования не существует:
 - а. логического;
 - б. структурного;
 - в. модульного;
 - г. объектно-ориентированного
- П23. Какой этап выполняется раньше остальных:
 - а. разработка алгоритма;
 - б. выбор языка программирования;
 - в. написание исходного кода;
 - г. компиляция
- П24. Наличие комментариев позволяет:
 - а. быстрее писать программы;
 - б. быстрее выполнять программы;
 - в. быстрее найти ошибки в программе;
 - г. быстрее произвести описание структуры программы
- П25. Что определяет выбор языка программирования:
 - а. область приложения;
 - б. знание языка;
 - в. наличие дополнительных библиотек;
 - г. особенности структуры
- П26. Для каких задач характерно использование большого количества исходных данных, выполнение операций поиска, группировки:
 - а. для экономических задач;
 - б. для системных задач;
 - в. для инженерных задач;
 - г. для математический
- П27. На каком этапе производится выбор языка программирования:
 - а. проектирование;
 - б. программирование;
 - в. отладка;
 - г. тестирование.
- П28. Когда приступают к тестированию программы:
 - а. когда программа уже закончена;
 - б. после постановки задачи;
 - в. на этапе программирования;
 - г. на этапе проектирования;
- П29. Одним из методов автоматизации программирования является:
 - а. структурное программирование;

- б. модульное программирование;
 - в. визуальное программирование;
 - г. объектно-ориентированное программирование.
- П30. Критерием оптимизации программы является:
- а. быстродействие или размер программы;
 - б. быстродействие и размер программы;
 - в. надежность или эффективность;
 - г. надежность и эффективность

Критерии оценки для проведения экзамена по дисциплине (тестирование)

Шкала оценивания результатов промежуточной аттестации и критерии выставления оценок.

Оценка	Уровень сформированности компетенции	Критерии
«Отлично» («зачтено»)	Высокий	Выполнено более 80% заданий предложенного теста
«Хорошо» («зачтено»)	Хороший	Выполнено 60-80% заданий предложенного теста
«Удовлетворительно» («зачтено»)	Достаточный	Выполнено 35-60% заданий предложенного теста
«Неудовлетворительно» («не зачтено»)	Недостаточный	Выполнено менее 35% заданий предложенного теста

Обновление оценочных материалов дисциплины

обсуждены и рекомендованы к утверждению решением кафедры информационных технологий и естественно-научных дисциплин от ____ 20__ г., протокол № ____

одобрены Научно-методическим советом университета от ____ 20__ г., протокол № ____