

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Погребная Ярослава Адольфовна
Должность: Директор
Дата подписания: 27.01.2025 14:40:14
Уникальный программный ключ:
df3b41101d3b2b77a07bf7ecfceb4c437367e6f2

**Частное образовательное учреждение
профессионального образования
«Налоговый колледж»**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

**ПМ.01 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ
КОМПЬЮТЕРНЫХ СИСТЕМ**

Специальность 09.02.07 Информационные системы и программирование

Квалификация выпускника: программист

Образовательная программа на базе среднего общего образования
Образовательная программа на базе основного общего образования

Формы обучения: очная

Москва 2024

Фонд оценочных средств профессионального модуля составлен на основании Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Минобрнауки России от 09.12.2016 № 1547 (с изменениями и дополнениями)

Фонд оценочных средств обсужден на заседании ПЦК общепрофессиональных и профессиональных дисциплин

Протокол № 5 от 24.05.2024

СОГЛАСОВАНО

Заместитель директора по УМР

Ложникова Т.В.

24.05.2024

СОДЕРЖАНИЕ

1.	ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ	4
2.	РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ	5
3.	ОЦЕНКА ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ	10
3.1	Контроль и оценка освоения профессионального модуля по разделам и темам	10
3.2	МДК.01.01 Разработка программных модулей	12
3.3	МДК.01.02 Поддержка и тестирование программных модулей	34
3.4	МДК.01.03 Разработка мобильных приложений	47
3.5	МДК.01.04 Системное программирование	56
4.	КРИТЕРИИ ОЦЕНКИ	59

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы профессионального модуля ПМ.01 Разработка модулей программного обеспечения для компьютерных систем и предназначен для контроля и оценки образовательных достижений, обучающихся (в т.ч. по самостоятельной работе студентов, далее – СРС), освоивших программу данного модуля.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям ФГОС СПО по специальности 09.02.07 Информационные системы и программирование (квалификация - программист).

Рабочей программой профессионального модуля ПМ.01 Разработка модулей программного обеспечения для компьютерных систем предусмотрено формирование следующих компетенций:

- 1) ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
- 2) ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;
- 3) ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях;
- 4) ОК 04. Эффективно взаимодействовать и работать в коллективе и команде;
- 5) ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
- 6) ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках;
- 7) ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;
- 8) ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием;
- 9) ПК.1.3. Выполнять отладку программных модулей с использованием специализированных программных средств;
- 10) ПК 1.4. Выполнять тестирование программных модулей;
- 11) ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода;
- 12) ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

Формой промежуточной аттестации по профессиональному модулю является экзамен (квалификационный).

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

В результате аттестации по профессиональному модулю осуществляется комплексная проверка следующих знаний, умений, практического опыта, а также динамика формирования общих и профессиональных компетенций.

Результаты обучения: знания, умения, практический опыт	Формируемые компетенции
Знать:	ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам
31 – актуальный профессиональный и социальный контекст, в котором приходится работать и жить;	
32 – основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте;	
33 – алгоритмы выполнения работ в профессиональной и смежных областях;	
34 – методы работы в профессиональной и смежных сферах;	
35 – структуру плана для решения задач;	
36 – порядок оценки результатов решения задач профессиональной деятельности.	
Уметь:	
У1 – распознавать задачу и/или проблему в профессиональном и/или социальном контексте;	
У2 – анализировать задачу и/или проблему и выделять её составные части;	
У3 – определять этапы решения задачи;	
У4 – выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы;	
У5 – составлять план действия;	
У6 – определять необходимые ресурсы;	
У7 – владеть актуальными методами работы в профессиональной и смежных сферах;	
У8 – реализовывать составленный план;	
У9 – оценивать результат и последствия своих действий (самостоятельно или с помощью наставника).	
Знать:	ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
31 – номенклатуру информационных источников, применяемых в профессиональной деятельности;	
32 – приемы структурирования информации;	
33 – формат оформления результатов поиска информации, современные средства и устройства информатизации;	
34 – порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств.	
Уметь:	
У1 – определять задачи для поиска информации;	
У2 – определять необходимые источники информации;	
У3 – планировать процесс поиска;	
У4 – структурировать получаемую информацию;	
У5 – выделять наиболее значимое в перечне информации;	

У6 – оценивать практическую значимость результатов поиска;	
У7 – оформлять результаты поиска, применять средства информационных технологий для решения профессиональных задач;	
У8 – использовать современное программное обеспечение;	
У9 – использовать различные цифровые средства для решения профессиональных задач.	
Знать:	
З1 – содержание актуальной нормативно-правовой документации;	
З2 – современная научная и профессиональная терминология;	
З3 – возможные траектории профессионального развития и самообразования;	
З4 – основы предпринимательской деятельности;	
З5 – основы финансовой грамотности;	
З6 – правила разработки бизнес-планов;	
З7 – порядок выстраивания презентации;	
З8 – кредитные банковские продукты.	
Уметь:	
У1 – определять актуальность нормативно-правовой документации в профессиональной деятельности;	
У2 – применять современную научную профессиональную терминологию;	
У3 – определять и выстраивать траектории профессионального развития и самообразования;	
У4 – выявлять достоинства и недостатки коммерческой идеи;	
У5 – презентовать идеи открытия собственного дела в профессиональной деятельности;	
У6 – оформлять бизнес-план;	
У7 – рассчитывать размеры выплат по процентным ставкам кредитования;	
У8 – определять инвестиционную привлекательность коммерческих идей в рамках профессиональной деятельности;	
У9 – презентовать бизнес-идею;	
У10 – определять источники финансирования.	
Знать:	
З1 – психологические основы деятельности коллектива, психологические особенности личности;	
З2 – основы проектной деятельности.	
Уметь:	
У1 – организовывать работу коллектива и команды;	
У2 – взаимодействовать с коллегами, руководством, клиентами в ходе профессиональной деятельности.	
Знать:	
З1 – особенности социального и культурного контекста;	
З2 – правила оформления документов и построения устных сообщений.	
Уметь:	
	ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях
	ОК 04. Эффективно взаимодействовать и работать в коллективе и команде
	ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей

У1 – грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, проявлять толерантность в рабочем коллективе.	социального и культурного контекста
Знать:	ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках
З1 – правила построения простых и сложных предложений на профессиональные темы;	
З2 – основные общеупотребительные глаголы (бытовая и профессиональная лексика);	
З3 – лексический минимум, относящийся к описанию предметов, средств и процессов профессиональной деятельности;	
З4 – особенности произношения;	
З5 – правила чтения текстов профессиональной направленности.	
Уметь:	
У1 – понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы;	
У2 – участвовать в диалогах на знакомые общие и профессиональные темы;	
У3 – строить простые высказывания о себе и о своей профессиональной деятельности;	
У4 – кратко обосновывать и объяснять свои действия (текущие и планируемые);	
У5 – писать простые связные сообщения на знакомые или интересующие профессиональные темы.	
Знать:	ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
З1 – основные этапы разработки программного обеспечения;	
З2 – основные принципы технологии структурного и объектно-ориентированного программирования;	
З3 – актуальную нормативно-правовую базу в области документирования алгоритмов.	
Уметь:	
У1 – формировать алгоритмы разработки программных модулей в соответствии с техническим заданием;	
У2 – оформлять документацию на программные средства;	
У3 – оценивать сложность алгоритма.	
Иметь практический опыт в:	ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
П1 – разработке алгоритма решения поставленной задачи и реализации его средствами автоматизированного проектирования.	
Знать:	ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
З1 – основные этапы разработки программного обеспечения;	
З2 – основные принципы технологии структурного и объектно-ориентированного программирования;	
З3 – API современных мобильных операционных систем.	
Уметь:	
У1 – создавать программу по разработанному алгоритму	

как отдельный модуль;	
У2 – оформлять документацию на программные средства;	
У3 – осуществлять разработку кода программного модуля на языках низкого уровня и высокого уровней в том числе для мобильных платформ.	
Иметь практический опыт в:	
П1 – разработке кода программного продукта на основе готовой спецификации на уровне модуля;	
П2 – разработке мобильных приложений.	
Знать:	
З1 – основные принципы отладки и тестирования программных продуктов;	
З2 – инструментарий отладки программных продуктов.	
Уметь:	
У1 – выполнять отладку и тестирование программы на уровне модуля;	
У2 – оформлять документацию на программные средства;	
У3 – применять инструментальные средства отладки программного обеспечения.	
Иметь практический опыт в:	
П1 – использовании инструментальных средств на этапе отладки программного продукта;	
П2 – проведении тестирования программного модуля по определенному сценарию.	
Знать:	
З1 – основные виды и принципы тестирования программных продуктов.	
Уметь:	
У1 – выполнять отладку и тестирование программы на уровне модуля;	
У2 – оформлять документацию на программные средства.	
Иметь практический опыт в:	
П1 – проведении тестирования программного модуля по определенному сценарию;	
П2 – использования инструментальных средства на этапе тестирования программного продукта.	
Знать:	
З1 – способы оптимизации и приемы рефакторинга;	
З2 – инструментальные средства анализа алгоритма;	
З3 – методы организации рефакторинга и оптимизации кода;	
З4 – принципы работы с системой контроля версий.	
Уметь:	
У1 – выполнять оптимизацию и рефакторинг программного кода;	
У2 – работать с системой контроля версий.	
Иметь практический опыт в:	
П1 – анализе алгоритмов, в том числе с применением инструментальных средств;	
П2 – осуществлении рефакторинга и оптимизации программного кода.	
	ПК.1.3. Выполнять отладку программных модулей с использованием специализированных программных средств
	ПК 1.4. Выполнять тестирование программных модулей
	ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода

Знать:	ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ
З1 – основные этапы разработки программного обеспечения;	
З2 – основные принципы технологии структурного и объектно-ориентированного программирования.	
Уметь:	
У1 – осуществлять разработку кода программного модуля на современных языках программирования;	
У2 – оформлять документацию на программные средства.	
Иметь практический опыт в:	
П1 – разработке мобильных приложений.	

3. ОЦЕНКА ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

3.1. Контроль и оценка освоения профессионального модуля по разделам и темам

Предметом оценки служат знания, умения и практический опыт, предусмотренные ФГОС СПО, направленные на формирование общих и профессиональных компетенций.

Элемент профессионального модуля	Формы и методы контроля			
	Текущий контроль		Промежуточная аттестация	
	Форма контроля	Проверяемые компетенции/ знания/ умения/ практический опыт	Форма контроля	Проверяемые компетенции/ знания/ умения/ практический опыт
МДК.01.01 Разработка программных модулей				
Раздел 1.1. Разработка программных модулей	Практические работы; Контрольные работы; Тестовые задания	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.1: 31-33, У1-У3, П1; ПК 1.2: 31-33, У1-У3, П1, П2.	Контрольная работа/ Дифференцированный зачет/ Экзамен	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.1: 31-33, У1-У3, П1; ПК 1.2: 31-33, У1-У3, П1, П2.
Тема 1.1.1. Жизненный цикл ПО				
Тема 1.1.2. Структурное программирование				
Тема 1.1.3. Объектно-ориентированное программирование				
Тема 1.1.4. Паттерны проектирования				
Тема 1.1.5. Событийно-управляемое программирование				
Тема 1.1.6. Оптимизация и рефакторинг кода				
Тема 1.1.7. Разработка пользовательского интерфейса				
Тема 1.1.8. Основы ADO.Net				
МДК.01.02 Поддержка и тестирование программных модулей				
Раздел 1.2. Поддержка и тестирование программных модулей	Практические работы; Контрольные работы; Тестовые задания	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.3: 31, 32, У1-У3, П1, П2;	Контрольная работа/ Дифференцированный зачет/ Экзамен	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.3: 31, 32, У1-У3, П1, П2;
Тема 1.2.1. Отладка и тестирование программного обеспечения				
Тема 1.2.2. Документирование				

		ПК 1.4: 31, У1, У2, П1, П2; ПК 1.5: 31-34, У1, У2, П1, П2.		ПК 1.4: 31, У1, У2, П1, П2; ПК 1.5: 31-34, У1, У2, П1, П2.
МДК.01.03 Разработка мобильных приложений				
Раздел 1.3. Разработка мобильных приложений	Практические и лабораторные работы; Контрольные работы; Тестовые задания	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.2: 31-33, У1-У3, П1, П2; ПК 1.6: 31, 32 У1, У2, П1.	Контрольная работа/ Дифференцированный зачет	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.2: 31-33, У1-У3, П1, П2; ПК 1.6: 31, 32 У1, У2, П1.
Тема 1.3.1. Основные платформы и языки разработки мобильных приложений				
Тема 1.3.2. Создание и тестирование модулей для мобильных приложений				
МДК.01.04 Системное программирование				
Раздел 1.4. Системное программирование	Практические работы; Контрольные работы; Тестовые задания	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.2: 31-33, У1-У3, П1, П2; ПК 1.3: 31, 32, У1-У3, П1, П2.	Дифференцированный зачет/ Экзамен	ОК 01: 31-36, У1-У9; ОК 02: 31-34, У1-У9; ОК 03: 31-38, У1-У10; ОК 04: 31, 32, У1, У2; ОК 05: 31, 32, У1; ОК 09: 31-35, У1-У5; ПК 1.2: 31-33, У1-У3, П1, П2; ПК 1.3: 31, 32, У1-У3, П1, П2.
Тема 1.4.1. Программирование на языке низкого уровня				

3.2. МДК.01.01. Разработка программных модулей

ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ РАЗДЕЛ 1. РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ

Тема 1.1.1. Жизненный цикл ПО

Вопросы с одним вариантом правильного ответа

1. С какого момента появляется жизненный цикл?

- а) С момента осознания появления, данного ПО;
- б) С момента его полного выхода из употребления.

2. Стадии жизненного цикла?

- а) Определения требований и спецификаций; проектирования; программирования и сопровождения;
- б) Надежность; технологичность; правильность.

3. Сколько стадий в ЖЦ программного обеспечения?

- а) 4;
- б) 5;
- в) 6.

4. Что делает программный–аналитик?

- а) анализирует и проектирует комплекс взаимосвязанных программ;
- б) разработчик формальных постановок задач, требующих реализации на ЭВМ.

5. Кто является разработчиком формальных постановок задач, требующих реализацию на ЭВМ?

- а) Программист–аналитик;
- б) Инженер;
- в) Постановщик задач.

6. Что определяют функциональные спецификации?

- а) функции, которые должно выполнять ПО, т.е. в них определяется, что надо делать системе, а не то, как это делать;
- б) Точность не позволяет различных толкований.

7. Что такое цель этапа?

- а) иерархическое разбиение сложных задач создания ПО на подзадачи меньшей сложности;
- б) Результатом работы на этом этапе являются спецификации на отдельные модули, дальнейшая декомпозиция которых нецелесообразна.

8. Для чего предназначен маркетинг?

- а) Результатом работы на этом этапе являются спецификации на отдельные модули, дальнейшая декомпозиция которых нецелесообразна;
- б) Изучение также существующие аналоги и продукты- конкуренты.

9. Чем характеризуется каскадная модель?

- а) характеризуется последовательным выполнением входящих в ее состав этапов, окончанием каждого предыдущего этапа до начала последующего, отсутствием временного перекрытия этапов (последующий этап не начнется, пока не завершится предыдущий), отсутствием (или определенным ограничением) возврата к предыдущим этапам. Выявление и устранение ошибок в каскадной модели производится только на этапе тестирования;
- б) Основной ее особенностью является наличие обратных связей между этапами, проведением проверок и корректировок проектируемого программного обеспечения на каждой стадии разработки

10. Кто такой администратор сети?

- а) человек, который обеспечивает организационную поддержку работы локальной сети;
- б) Основной потребитель программ.

11. Основным потребителем программ является?

- а) конечный пользователь, который, как правило, не является специалистом в области программирования;
- б) Он не программирует, а используют готовые программы для обеспечения работы на ЭВМ.

12. Наличие обратных связей между этапами жизненного цикла, определяющих возможность проведения проверок и корректировок проектируемого ПС на каждой стадии разработки, является основной особенностью ...

- а) Каскадной модели жизненного цикла ПО;
- б) Итерационной модели жизненного цикла ПО;
- в) Спиральной модели жизненного цикла ПО;
- г) Модели жизненного цикла UML.

13. Выявлением и устранением ошибок только на стадии тестирования, которая может растянуться во времени или вообще никогда не завершиться, характеризуется ...

- а) каскадной модели жизненного цикла
- б) итерационной модели жизненного цикла
- в) спиральной модели жизненного цикла ПО
- г) модели жизненного цикла UML

Вопросы с несколькими вариантами правильного ответа

1. Какие требования устанавливаются на первой стадии?

- а) по надежности;
- б) технологичности;
- в) правильности;
- г) универсальности;
- в) эффективности;
- д) информационной согласованности;

2. Перечислите значения спецификаций.

- а) Спецификации являются заданием на разработку ПО и их
- б) выполнение - закон для разработчика.
- в) Спецификации используются для проверки готовности ПО.
- г) Спецификации являются неотъемлемой частью программной документации, облегчают сопровождение и модификацию ПО.

3. Что происходит на второй стадии?

- а) Формируется структура ПО и разрабатываются алгоритмы, задаваемые спецификациями;
- б) Устанавливается состав модулей с разделением их на иерархические уровни на основе изучения схем алгоритмов;
- в) Выбирается структура информационных массивов;
- г) Фиксируются межмодульные интерфейсы;

Тема 1.1.2. Структурное программирование

Вопросы с одним вариантом правильного ответа

1. Алгоритм вычисления значений функций $F(w)$ и $Q(w)$, где w — натуральное число, задан следующими соотношениями:

- а) $F(1) = 1; Q(1) = 1;$
- б) $F(w) = F(w-1) - 2 * Q(w-1)$ при $w > 1$
- в) $Q(w) = Q(w-1) - 2 * F(w-1)$ при $w > 1.$

Чему равно значение функции $F(5)$ $Q(5)$?

- а) 62
- б) -14
- в) -12
- д) 31

2. Дан рекурсивный алгоритм:

```
a)procedure (n:integer);
begin writeln('*');
if n > 0 then begin
F(n-2);
F(n div 2);
F(n div 2);
end
end;
```

Сколько символов «звездочка» будет напечатано на экране при выполнении вызова $F(5)$?

- а) 12
- б) 23
- в) 34
- д) 45

3. Основная идея структурного программирования состоит в том, что

- а) При написании программ не используются подпрограммы; используется инкапсуляция и наследование объектов;
- б) Для реализации алгоритма рекомендуется использовать операторы безусловного перехода;
- в) Для написания программ используются только три типа операторов: линейный, ветвление, повторение цикла

Задачи, модельные ситуации или проблемы, ориентированные на профессиональную деятельность.

(Внимательно прочитайте текст задания, продумайте логику и полноту ответа. Запишите ответ, в случае расчетной задачи, запишите решение и ответ)

4. Дан рекурсивный алгоритм:

Найдите сумму чисел, которые будут выведены при вызове F(2).

```
procedure F(n: integer);
begin
  writeln(n);
  if n < 5 then begin
    F(n - 1);
    F(n - 2);
    F(n*3)
  end
end;
```

-

5. Дан рекурсивный алгоритм:

Чему будет равно значение, вычисленное алгоритмом при выполнении вызова F(5)?

```
function F(n: integer): integer;
begin
  if n > 2 then
    F := F(n - 1) * F(n - 2)
  else
    F := n;
end;
```

-

6. Ниже записаны две рекурсивные функции F и G:

Чему будет равно значение, вычисленное при выполнении вызова F(6)?

```
function F(n: integer): integer;
begin
  if n > 2 then
    F := F(n - 1) * G(n - 2)
  else
    F := n;
end;
function G(n: integer): integer;
begin
  if n > 2 then
    G := G(n - 1) * F(n - 2)
  else
    G := n - 1;
end;
```

-

Задания и контрольные вопросы по проверке результатов практических работ

Описание

Проверка полученных на уроках знаний и результат формируемых умений по **Теме Структурное программирование** осуществляется на практических работах. Результаты выполнения проверяется преподавателем, и оценка выставляется после ответов на контрольные вопросы практической работы).

Тема 1.1.3. Объектно-ориентированное программирование

1. Вопросы с одним вариантом правильного ответа

1. В основе концепции объектно-ориентированного программирования лежит понятие:

- a) Объекта
- b) Класса
- c) Инкапсуляции

2. Инкапсуляция – это:

- a) Свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствующейся функциональностью.
- b) Сущность в адресном пространстве вычислительной системы, появляющаяся при создании экземпляра класса или копирования прототипа (например, после запуска результатов компиляции и связывания исходного кода на выполнение)
- c) Свойство системы, позволяющее объединить данные и методы, работающие с ними в классе, и скрыть детали реализации от пользователя.

3. Способ выделить набор значимых характеристик объекта, исключая из рассмотрения незначимые.

- a) Полиморфизм
- b) Абстрагирование
- c) Прототип

4. Термин "наследование" обозначает, что...

- a) В производных классах присутствует часть состояния родительского класса
- b) Производные классы содержат поля и методы родительского
- c) Производные классы наследуют модификаторы доступа членов родительского класса

5. Способ защититься от использования объектов одного класса вместо другого, или по крайней мере управлять таким использованием – это:

- a) Типизация
- b) Наследование
- c) Полиморфизм

6. В каких случаях вызывается деструктор:

- a) Создание объекта

- b) Удаление объекта
- c) Редактирование объекта

2. Вопросы с несколькими вариантами правильных ответов

7. В каких отношениях может находиться один класс с другим:

- a) Отношение наследования
- b) Отношение включения
- c) Отношение использования

3.Задания

На соответствия понятий и содержания, задания на дополнения, задания на сопоставление, задания на установление правильной последовательности (Установите соответствие, добавив в свободные поля к цифрам буквы.)

8. Соотнесите понятия

A) Состояние объекта	1.некоторый объект или множество объектов
B) Поведение объекта	2.набор методов (программный код), оперирующих над состоянием объекта
C) Значение атрибута объекта	3.набор значений его атрибутов

A-3

B-2

C-1

9. Соотнести понятия спецификаторов:

Спецификаторы	Определения
A) private	1. защищенный, разрешено обращаться как с текущего класса, так и с классов наследников
B) protected	2. закрытый, то есть к нему можно обращаться только в текущем классе
C) public	3. общедоступный, разрешено обращаться из любого места программы

A-2

B-2

C-3

10. Соотнесите определения

A) Caption	1.Определяет общий вид окна и операции с ним, которые разрешается выполнять пользователю
Б) Icon	2. Определяет заголовок окна приложения
Б) Icon	3. Определяет значок, который будет использоваться в строке заголовка формы при выполнении программы

A-2

B-3

C-1

Тема 1.1.4. Паттерны проектирования

Вопросы с одним вариантом правильного ответа

1. К какому из видов относится шаблон Итератор?

- a) Порождающий
- b) Структурный
- c) Поведенческий
- d) Базовый

2. Гарантирует, что класс имеет только один экземпляр и представляет глобальную точку доступа к нему. О каком паттерне идет речь?

- a) Фасад
- b) Одиночка
- c) Фабричный метод
- d) Приспособленец

3. Разделяет создание сложного объекта и инициализацию его состояния так, что одинаковый процесс построения может создать объекты с разным состоянием. О каком паттерне идет речь?

- a) Строитель
- b) Мост
- c) Адаптер
- d) Компоновщик

4. Благодаря совместному использованию, поддерживает эффективную работу с большим количеством объектов. О каком паттерне идет речь?

- a) Приспособленец
- b) Одиночка
- c) Фасад
- d) Фабричный метод

5. Выберите верное утверждение

- a) Декоратор относится к порождающим паттернам
- b) Компоновщик относится к порождающим паттернам
- c) Мост относится к порождающим паттернам
- d) Строитель относится к порождающим паттернам

6. Предоставляет интерфейс для создания групп связанных или зависимых объектов, не указывая их конкретный класс. О каком паттерне идет речь?

- a) Стратегия
- b) Декоратор
- c) Абстрактная фабрика
- d) Состояние

7. Что такое зависимость?

- a) описывает связь "часть"- "целое", в которой "часть" может существовать от

"целого"

- b) Подвид агрегации, в которой "части" не могут существовать отдельно от "целого"
- c) Изменение в одной сущности может влиять на состояние или поведение другой сущности
- d) Отношение наследования или реализации интерфейса

Тема 1.1.5. Событийно -управляемое программирование

Вопросы с одним вариантом правильного ответа

1. Спецификация разрабатываемого программного обеспечения при использовании языка визуального моделирования UML объединяет несколько моделей. Определяет реальную организацию программных модулей в среде разработки ...

- a. модель использования
- b. модель реализации
- c. модель процессов
- d. модель развертывания

2. Спецификация разрабатываемого программного обеспечения при использовании языка визуального моделирования UML объединяет несколько моделей. Отображает организацию вычислений и позволяет оценить производительность, масштабируемость и надежность программного обеспечения ...

- a. модель использования
- b. модель реализации
- c. модель процессов
- d. модель развертывания

3. Спецификация разрабатываемого программного обеспечения при использовании языка визуального моделирования UML объединяет несколько моделей. Показывает, каким образом программные компоненты размещаются на конкретном оборудовании ...

- a. модель использования
- b. модель реализации
- c. модель процессов
- d. модель развертывания

4. Служит связующим звеном между процессами, т. е. позволяет описать порядок взаимодействия процессов ...

- a. нотация
- b. прецедент
- c. поток событий
- d. сценарий

Тема 1.1.6. Оптимизация и рефакторинг кода
Тема 1.1.7. Разработка пользовательского интерфейса.

1. Определите, о каком принципе пользовательского интерфейса идет речь:

«Пользователь должен получать сообщения о действиях системы и о важных событиях внутри нее. Сообщения должны быть информативными, краткими, однозначными и написанными на языке, понятном пользователю.»

- a) Принцип простоты
- b) Принцип структуризации
- c) Принцип обратной связи
- d) Принцип толерантности

2. Определите, о каком принципе пользовательского интерфейса идет речь:

«Все функции и данные, необходимые для решения определенной задачи, должны быть видны, когда пользователь пытается ее решить.»

- a) Принцип повторного использования
- b) Принцип видимости
- c) Принцип обратной связи
- d) Принцип простоты

3. Выберите описание правила эффективности:

a) «Система должна способствовать непрерывному росту знаний, умений и навыков пользователя и приспосабливаться к его меняющемуся опыту. Плохие результаты приносит предоставление только базовых возможностей или оставление начинающего пользователя наедине со сложным интерфейсом, которым уверенно пользуются эксперты.»

b) «Система не должна препятствовать эффективной работе опытных пользователей, работающих с ней долгое время. Очевидным примером нарушения этого правила является нацеленность системы только на новичков, использование средств, которые хорошо подходят для неопытного пользователя, ограничивая его в возможности сделать что-то не так, но неэффективны для эксперта, который и так знает, что и где ему нужно сделать.»

c) «Система должна быть настолько понятной, чтобы пользователь, никогда раньше не видевший ее, но хорошо разбирающийся в предметной области, мог без всякого обучения начать ее использовать. Это правило служит некоторым идеалом, к которому надо стремиться, поскольку на практике достичь такой степени доступности почти никогда не удается. Тем не менее, специалисты продолжают делать в этом направлении все возможное.»

2. Вопросы с несколькими вариантами правильных ответов

4. Отметьте только принципы использования цвета, которыми нужно руководствоваться при проектировании эргономичного интерфейса:

- a) для неактивных элементов лучше использовать бледные цвета;
- b) выровненный по правому краю текст труднее читать, чем равномерно распределенный текст с невыровненным правым полем если надо разделить данные, то цвета выбирают из различных частей спектра: красный, зеленый, синий — желтый, любой цвет — белый

- с) структура меню должна соответствовать структуре решаемой системой задачи; организация меню должна отразить наиболее эффективную последовательность шагов, ведущих к решению поставленной задачи
- д) при отображении состояния, как правило, красный означает опасность (стоп), зеленый — продолжение работы, желтый — цвет предупреждения

5. Отметьте только *принципы*, которыми необходимо руководствоваться при создании текстовых диалогов:

- а) для неактивных элементов лучше использовать бледные цвета;
- б) выровненный по правому краю текст труднее читать, чем равномерно распределенный текст с не выровненным правым полем
- в) текст в нижнем регистре читается приблизительно на 13% быстрее, чем текст, напечатанный полностью в верхнем регистре
- г) пункты меню должны быть краткими, грамматически правильными и соответствовать своему заголовку. Порядок пунктов меню выбирается в соответствии с соглашением, частотой и порядком использования, а также в зависимости от потребностей задачи или пользователя
- д) для группировки данных, объединения и подобия нужно использовать соседние цвета спектра: оранжевые—желтые, синие—фиолетовые

3.Задания

На соответствия понятий и содержания, задания на дополнения, задания на сопоставление, задания на установление правильной последовательности (Установите соответствие, добавив в свободные поля к цифрам буквы.)

6. Установите соответствие

А) размер (текста, символов)	1) эффективный метод для привлечения внимания к какой-либо части экрана
В) оттенение (различная текстура объектов)	2) обычно применяют увеличение выделенного объекта в 1,5 раза
С) окружение (подчеркивание, рамки, инвертированное изображение)	3) Очень эффективный способ, если не переусердствовать

A-2

B-2

C-3

7. Установите соответствие

а) форма (символ, шрифт, форма символа)	1) эффективный метод для привлечения внимания к какой-либо части экрана
б) размер (текста, символов)	2) используется для того, чтобы дифференцировать различные категории данных
с) оттенение (различная текстура объектов)	3) обычно применяют увеличение выделенного объекта в 1,5 раза

A-2

B-3

C-1

1. Метод WriteXml для набора данных записывает данные набора данных в XML-файл. WriteXml принимает параметр для различных типов назначений. Какой из перечисленных ниже типов является недопустимым для пункта назначения?

- A. Строка
- B. System.IO.Stream
- C. System.IO.TextStream
- D. System.IO.TextWriter

2. ADO.NET известен отключенным извлечением данных. Какой из перечисленных ниже механизмов не использует отключенный механизм при извлечении данных из базы данных?

- A. адаптер данных
- B. DataReader
- C. Набор данных
- D. Ничего из вышеперечисленного

3. Что из перечисленного не входило в число основных целей дизайна ADO NET?

- A. Обеспечить бесперебойную поддержку XML
- B. Для прямой поддержки <url>
- C. Обеспечить расширяемую и масштабируемую архитектуру доступа к данным для революционной многоуровневой модели программирования
- D. Расширить текущие возможности ADO

4. Наиболее распространенными в практике являются:

- A. распределенные базы данных;
- B. иерархические базы данных;
- C. сетевые базы данных; D.
- реляционные базы данных.

5. Без каких объектов не может существовать база данных:

- A. без модулей;
- B. без отчетов;
- C. без таблиц;
- D. без форм;
- E. без макросов;

Задачи, модельные ситуации или проблемы, ориентированные на профессиональную деятельность

Для групповых операций с файлами используются маски имён файлов. Маска представляет собой последовательность букв, цифр и прочих допустимых в именах файлов символов, в которых также могут встречаться следующие символы:

Символ «?» (вопросительный знак) означает ровно один произвольный символ.

Символ «*» (звёздочка) означает любую последовательность символов произвольной длины, в том числе «*» может задавать и пустую последовательность.

Определите, какое из указанных имён файлов удовлетворяет маске:

??pri*.?*

- 1) napri.q

- 2) pripri.txt
- 3) privet.doc
- 4) Зpriveta.c

Сколько записей удовлетворяют условию «Пол = 'ж' ИЛИ Физика = 79»?

Фамилия	Пол	Математика	История	Физика	Химия	Биология
Андреев	м	80	72	68	66	70
Борисов	м	75	88	69	61	69
Васильева	ж	85	77	73	79	74
Дмитриев	м	77	85	81	81	80
Егорова	ж	88	75	79	85	75
Захарова	ж	72	80	66	70	70

В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите ID дяди Керзона П. А. (дядя — это родной брат матери или отца).

ID	Фамилия_И.О	Пол
3272	Базилевич А. Б.	Ж
3228	Базилевич Б. Ф.	М
3299	Базилевич П. Б.	М
3278	Базилевич П. П.	М
3256	Базилевич Т. П.	Ж
3265	Керзон А. П.	Ж
3331	Керзон А. П.	М
3261	Керзон Л. А.	Ж
1217	Керзон П. А.	М
1202	Зельдович М. А.	Ж
3227	Витюк Д. А.	Ж
3240	Лемешко В. А.	Ж
3246	Месяц К. Г.	М
3387	Лукина Р. Г.	Ж
3293	Фокус П. А.	Ж
3332	Друк Г. Р.	Ж

ID_Родителя	ID_ребенка
3272	3272
3228	3299
3299	3272
3278	3299
3256	3240
3265	1202
3331	1217
3261	3356
1217	3378
1202	3356
3227	3378
3240	3240
3246	1202
3387	1217
3293	3261
3332	3293

ОЦЕНОЧНЫЕ СРЕДСТВА ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Задание 1. Что такое жизненный цикл программного обеспечения (ЖЦ ПО)?

Выберите правильный ответ:

- а) это последовательность этапов разработки программного продукта, начиная с его концепции и заканчивая прекращением использования;
- б) это последовательность этапов деятельности программиста, начиная с идеи концепции и заканчивая созданием проекта;
- в) это последовательности кода программы, начиная с разработки его структуры и тестированием ПО.

Задание 2. Каковы ключевые задачи этапа анализа требований? Выберите все правильные ответы:

- а) определение потребностей пользователей;
- б) сбор и документирование требований;
- в) анализ требований на полноту и целостность;
- г) сбор информации для митапа.

Задание 3. Укажите основные задачи этапа разработки программного обеспечения:

- а) контроль дисциплины сотрудников;
- б) составление тех. документации;
- в) написание и документирование кода;
- г) реализация функционала согласно требованиям и проекту;
- д) контроль качества кода.

Задание 4. Что такое структурное программирование? Выберите правильный ответ:

- а) это подход к программированию, в основе которого лежит представление программы в виде иерархической структуры блоков;
- б) это метод программирования, который предполагает использование последовательного программного кода;
- в) это подход к программированию, который предполагает использование структур в виде классов и объектов.

Задание 5. Выберите ответ, в котором перечислены основные принципы структурного программирования:

- а) материальность, ветвление, циклы;
- б) абстрактность, ветвление, циклы;
- в) последовательность, ветвление, циклы.

Задание 6. Выберите ответ, в котором перечислены преимущества использования структурного программирования:

- а) повышенная сложность кода, увеличение ошибок, усложнение отладки и сопровождения;
- б) повышенная читаемость кода, уменьшение количества ошибок, упрощение отладки и сопровождения;
- в) повышенная читаемость кода, незначительные ошибки, отсутствие необходимости отладки и сопровождения.

Задание 7. Выберите ответ, в котором правильно описаны отличия структурного программирования от объектно-ориентированного программирования:

- а) структурное программирование фокусируется на логике и последовательности выполнения кода, а объектно-ориентированное программирование акцентирует внимание на

объектах и их взаимодействии;

б) структурное программирование фокусируется на объектах и их взаимодействии, а объектно-ориентированное программирование акцентирует внимание на логике и последовательности выполнения кода;

в) структурное и объектно-ориентированное программирование различаются тем, что структурное разделяет программы на маленькие функции, а объектно-ориентированное объединяет данные и функции в объектах.

Задание 8. Какой тип управления потоками используется в структурном программировании? Выберите правильный ответ:

- а) последовательность.
- б) классы и объекты;
- в) ветвление (if-else);
- г) циклы (for, while).

Задание 9. Какие инструментальные средства используются для документирования алгоритмов программ? Выберите все правильные ответы:

- а) блок-схемы;
- б) псевдокод, комментарии в коде;
- в) диаграммы классов и последовательностей;
- г) спецблокноты;
- д) печатающиеся устройства.

Задание 10. Выберите варианты ответов, в которых указаны популярные инструменты для создания блок-схем:

- а) Microsoft Visio;
- б) ramus.com;
- в) Linux;
- г) draw.io.

Задание 11. Какие средства используются для оформления документации на сложные алгоритмы? Выберите правильный ответ:

- а) детализированные блок-схемы, псевдокод, комментарии в коде и текстовые описания алгоритмов;
- б) упрощенные блок-схемы, программный код, инструкция по эксплуатации;
- в) спецблокноты, печатающиеся устройства, черновики документов.

Задание 12. Какие существуют стандарты для программной документации? Выберите все правильные ответы:

- а) международные стандарты;
- б) национальный стандарт РФ/ГОСТ Р;
- в) спецификации для отдельных ЯП;
- г) программирование - свободная профессия, какие-либо стандарты не предусмотрены.

Задание 13. Что такое временная сложность алгоритма? Выберите правильный ответ:

- а) временная сложность алгоритма измеряет, как время выполнения алгоритма зависит от размера входных данных;
- б) временная сложность алгоритма измеряет, как структура алгоритма зависит от размера входных данных;
- в) временная сложность алгоритма – это количество операций, которые алгоритм выполняет в зависимости от размера входных данных.

Задание 14. Что такое пространственная сложность алгоритма? Выберите правильный ответ:

- а) пространственная сложность алгоритма оценивает объем памяти, необходимой алгоритму для выполнения, в зависимости от размера входных данных;
- б) пространственная сложность алгоритма оценивает сложность логистики, необходимой разработчику для доставки программного обеспечения заказчику;
- в) пространственная сложность алгоритма оценивает сложность иерархии компьютерной сети для достижения максимальной вычислительной мощности.

Задание 15. Какие существуют основные классы сложности алгоритмов? Выберите правильные ответы:

- а) $O(1)$ — постоянная сложность;
- б) $O(n)$ — линейная сложность;
- в) $O(n^2)$ — квадратичная сложность;
- г) $O(n^3)$ - кубическая сложность.

Задание 16. Как определить временную сложность алгоритма? Выберите правильный ответ:

- а) временная сложность определяется путем анализа количества основных операций (например, сравнения, присваивания) в зависимости от размера входных данных;
- б) временная сложность алгоритма оценивает объем памяти, необходимой алгоритму для выполнения, в зависимости от размера входных данных;
- в) временная сложность определяется путем подсчета времени, необходимого для выполнения основных операций (например, сравнения, присваивания).

Задание 17. Что такое неразрешимая задача в теории алгоритмов? Выберите правильный ответ:

- а) это задача, для которой не существует алгоритма, способного её решить за конечное время для всех возможных входных данных;
- б) это задача, для которой не существует известного решения за любое конечное время для всех возможных входных данных;
- в) это задача, для которой не существует алгоритма, способного её решить за любое бесконечное время при всех неизвестных входных данных.

Задание 18. Выберите ответ, в котором указаны примеры алгоритмов с разной временной сложностью:

- а) $O(1)$: доступ к элементу массива по индексу;
- б) $O(n)$: линейный поиск;
- в) $O(n^2)$: сортировка пузырьком;
- г) $O(n^3)$ кубическая сортировка.

Задание 19. Чем отличается экспоненциальная сложность от полиномиальной? Выберите правильный ответ:

- а) полиномиальная сложность (например, $O(n^2)$) растет медленнее, чем экспоненциальная (например, $O(2^n)$), и является более предпочтительной для масштабируемых решений;
- б) полиномиальная сложность (например, $O(n^2)$) растет быстрее, чем экспоненциальная (например, $O(2^n)$), и является менее предпочтительной для масштабируемых решений;
- в) полиномиальная сложность (например, $O(n^2)$) растет медленнее, чем экспоненциальная (например, $O(2^n)$), и является менее предпочтительной для масштабируемых решений.

Задание 20. Какие методы используются для оценки сложности алгоритмов? Выберите все правильные ответы:

- а) анализ худшего случая, среднего случая и лучшего случая;

- б) использование асимптотических обозначений;
- в) экспериментальное тестирование;
- г) алгоритмическое тестирование.

Задание 21. Что такое перегрузка методов и зачем она используется? Выберите правильный ответ:

- а) перегрузка методов — это способность одного класса иметь несколько методов с одним именем, но разными параметрами. Это используется для улучшения читаемости и удобства использования кода;
- б) перегрузка методов — это способность нескольких классов иметь несколько методов с одним именем, но разными параметрами. Это используется для улучшения читаемости и удобства использования кода;
- в) перегрузка методов — это способность одного нескольких классов иметь один метод с одним именем и одним параметром.

Задание 22. Какие требования должны быть выполнены для перегрузки метода? Выберите правильный ответ:

- а) методы должны отличаться количеством и/или типами параметров. Возвращаемый тип значения не влияет на перегрузку;
- б) методы не должны отличаться количеством и/или типами параметров. Возвращаемый тип значения влияет на перегрузку;
- в) какие-либо требования отсутствуют.

Задание 23. Что такое делегат в ООП? Выберите правильный ответ:

- а) это экземпляр класса или его копия, которая находится в памяти компьютера;
- б) это класс, который позволяет хранить в себе ссылку на метод с определённой сигнатурой (порядком и типами принимаемых и типом возвращаемого значений);
- в) участник конференции программистов.

Задание 24. Что такое регулярные выражения и для чего они используются? Выберите правильный ответ:

- а) регулярные выражения — это мощный инструмент для поиска и обработки текста, позволяющий описывать шаблоны символов;
- б) регулярные выражения используются для поиска, замены и валидации текста.

Задание 25. Какие основные метасимволы используются в регулярных выражениях? Выберите все правильные ответы:

- а) ^ (начало строки);
- б) \$ (конец строки);
- в) * (ноль или более повторений);
- г) (одно или более повторений);
- д) ? (ноль или одно повторение);
- е) [] (символьный класс);
- ж) () (группировка);
- з) | (или).

Задание 26. Выберите все ответы, в которых указаны виды коллекций:

- а) списки;
- б) множества;
- в) словари;
- г) очереди;
- д) стеки;
- е) библиотеки.

Задание 27. Какие основные операции можно выполнять со списками? Выберите все правильные ответы:

- а) добавление элемента;
- б) удаление элемента;
- в) поиск элемента;
- г) сортировка;
- д) игнорирование элемента.

Задание 28. Что такое интерфейс в объектно-ориентированном программировании? Выберите верный ответ:

- а) это абстрактный тип, который содержит объявления методов, но не их реализацию;
- б) это метод, который выводит изображение на монитор;
- в) это некоторая сущность в цифровом пространстве, обладающая определённым состоянием и поведением, имеющая определённые свойства (поля) и операции над ними (методы).

Задание 29. Какие виды интерфейсов существуют? Выберите правильный ответ:

- а) аппаратные (в том числе для периферийных устройств);
- б) программные (software interfaces);
- в) пользовательские (user interfaces).

Задание 30. Как объявить интерфейс в языке программирования Java или C#? Выберите правильный ответ:

- а) интерфейс объявляется с использованием ключевого слова interface;
- б) формальное понятие интерфейсов отсутствует, объявлять этот тип данных нет необходимости;
- в) интерфейс объявляется по усмотрению пользователя.

Задание 31. Какие особенности наследования интерфейсов существуют в программировании? Выберите правильный ответ:

- а) интерфейсы могут наследовать другие интерфейсы, а классы могут реализовывать несколько интерфейсов. Реализация интерфейса обязательна для всех методов, определенных в интерфейсе;
- в) интерфейсы не могут наследовать другие интерфейсы, также как и классы не могут реализовывать несколько интерфейсов. Реализация интерфейса необязательна для всех методов, определенных в интерфейсе.

Задание 32(1). Сопоставьте процессы с этапами внедрения и сопровождения ПО:

1. Внедрение	а) поддержка и обновление ПО, исправление ошибок, адаптация к новым требованиям;
2. Сопровождение	б) установка и настройка ПО, обучение пользователей.

Задание 33(2). Сопоставьте понятия и, соответствующие им определения:

1. Инкапсуляция	а) принцип выделения значимых характеристик объекта и игнорирования несущественных деталей, помогает упростить сложные системы и позволяет фокусироваться на логике;
-----------------	--

2. Наследование	б) принцип, позволяющий одному классу (дочернему) наследовать свойства и методы другого класса (родительского), применяется для повторного использования кода и создания иерархий классов;
3. Полиморфизм	в) позволяет объектам разных классов обрабатывать данные по-разному через один и тот же интерфейс;
4. Абстракция	г) это принцип ООП, который предполагает скрытие внутреннего состояния объекта и предоставление доступа к нему только через методы, реализуется с помощью модификаторов доступа (например, private, public).

Задание 34(3). Сопоставьте структуры и классы с их характеристиками:

1. Структуры имеют	а) значения по типу (value type);
2. Классы имеют	б) ссылочные типы (reference type).

Задание 35(1). Укажите в правильной последовательности основные этапы ЖЦ ПО:

- а) разработка;
- б) проектирование;
- в) анализ требований;
- г) сопровождение;
- д) внедрение;
- е) тестирование.

Задание 1. Что является ключевым результатом этапа проектирования в ЖЦ ПО?

Задание 2. Какую задачу решает этап сопровождения ПО в ЖЦ?

Задание 3. Как называется графическое представление алгоритма или процесса?

Задание 4. К какому типу алгоритмов относится сортировка пузырьком?

Задание 5. Как расшифровывается аббревиатура ЕСПД?

Задание 6. К категории каких задач относится алгоритм, который для любой программы и любого входного набора данных сможет определить, завершится ли выполнение этой программы за конечное время или нет?

Задание 7. Как называется реализация (экземпляр) класса, которая содержит конкретные значения атрибутов и методов, определенных в классе. Каждый объект обладает своим собственным состоянием и поведением, которое определяется через данные и методы этого класса?

Задание 8. Как называется механизм объектно-ориентированного программирования, который позволяет создавать новые классы на основе уже существующих?

3.4. МДК.01.02 Поддержка и тестирование программных модулей

ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕГО КОНТРОЛЯ

Раздел 1.2. Поддержка и тестирование программных модулей

Тема 1.2.1. Отладка и тестирование программного обеспечения

Вопросы с выбором одного варианта правильного ответа

- 1. Какие существуют особенности управляющего графа программы в случае объектно-ориентированного программирования (ООП)?**
 - a. она становится неприменимой
 - b. она требует адаптации по обработке сообщений
 - c. она требует описания поведения программы
 - d. она требует описания не только структуры, но и поведения программы

- 2. Какие существуют особенности интеграционного тестирования для ООП?**
 - a. тестирование дерева классов
 - b. тестирование последовательностей прямых вызовов методов с помощью Р-путей
 - c. тестирование последовательностей обработки сообщений с помощью ММ-путей
 - d. тестирование исключительно цепочек вызовов процедур программ

- 3. Какие этапы включает методика ООП при тестировании программного комплекса?**
 - a. тестирование методов каждого класса программного комплекса
 - b. тестирование отношений между классами с помощью тестов на основе Р-путей или ММ-путей
 - c. тестирование взаимодействия модулей по всей иерархии комплекса

- 4. Какие этапы методики тестирования используются в ООП?**
 - a. тестирование методов каждого класса программы
 - b. тестирование методов класса, входящих в его контекст
 - c. тестирование дерева классов программного проекта, включающего оттестированный класс

- 5. Какие этапы методов тестирования класса используются в ООП?**
 - a. тестирование класса как модуля по выбранному критерию
 - b. тестирование класса как иерархической структуры
 - c. тестирование классов, входящих в модель проекта

- 6. Что такое управляющий граф программы (УГП)?**
 - a. множество операторов программы
 - b. граф, вершины которого кодируют операторы программы, а дуги - управления (порядок исполнения) операторов
 - c. множество операторов управления

- 7. Что такое путь в УГП?**
 - a. последовательность вершин и дуг управляющего графа программы с фиксированными начальной и конечной вершиной
 - b. последовательность ветвей управляющего графа программы с фиксированными начальной вершиной первой ветви и конечной вершиной последней ветви пути
 - c. множество связанных дуг управляющего графа программы

- 8. Что такое ветвь управляющего графа программы?**
- a) последовательность вершин и дуг управляющего графа программы с фиксированными начальной и конечной вершиной, которые кодируют либо условные операторы, либо первый и последний операторы управляющего графа программы соответственно
 - b. часть пути, в котором все внутренние вершины кодируют линейные операторы
 - c. начальная и конечная вершина пути
- 9. Можно ли гарантировать остановку программы на любом тесте?**
- a. в общем случае нет
 - b. возможно в частных случаях
 - c. задача в общей постановке алгоритмически неразрешима
- 10. Какие задачи у модульного тестирования?**
- a. выявление локальных ошибок реализации алгоритмов модулей
 - b. выявление ошибок при вызове модулей
 - c. выявление ошибок взаимодействия модуля с окружением
- 11. На основе каких принципов строятся тесты для модульного тестирования?**
- a. анализ потоков управления модуля
 - b. анализ потоков данных модуля
 - c. анализ покрытия в соответствии с заданными критериями C0, C1, C2
- 12. Каковы фазы процесса построения тестовых путей?**
- a. построение управляющего графа программы
 - b. выбор тестовых путей
 - c. генерация тестов, соответствующих выбранным тестовым путям
- 13. Какие существуют методы построения тестовых путей?**
- a. статические
 - b. динамические
 - c. методы реализуемых путей
- 14. Как реализуются динамические методы построения тестовых путей?**
- d. наращивание начальных отрезков реализованных путей продолжающими их фрагментами, чтобы увеличить покрытие
 - e. построение пути методом удлинения за счет добавления дуг
 - f. поиск всех реализуемых путей
- 15. Какие существуют разновидности интеграционного тестирования?**
- a. монолитное тестирование
 - b. нисходящее тестирование
 - c. восходящее тестирование
 - d. Регрессионное тестирование
- 16. Каковы особенности нисходящего тестирования?**
- a. необходимость разработки заглушек
 - b. необходимость разработки среды управления очередностью вызовов модулей
 - c. параллельная разработка эффективных модулей
- 17. Каковы особенности восходящего тестирования?**
- a. минимизация разработки заглушек
 - b. запаздывание в проверке функциональности реализуемого приложения
 - c. необходимость разработки среды управления очередностью вызовов модулей

- 18. В чем заключаются особенности интеграционного тестирования для процедурного программирования?**
- g. тестирование программных комплексов, заданных в виде иерархических структур модулей
 - h. использование диаграмм потока управления в качестве модели тестируемого комплекса
 - i. контроль соответствия спецификациям параметров модулей и межмодульных связей
 - j. контроль наследования
- 19. При создании очередной версии программы была добавлена функция А, функция D была удалена, функция С – изменена. К какой группе относится тест, покрывающий только функцию А?**
- a. тесты, пригодные для повторного использования
 - b. тесты, требующие повторного запуска
 - c. устаревшие тесты
 - d. новые тесты
- 20. Возможно ли тестирование программы на всех допустимых значениях параметров?**
- a. никогда
 - b. да, всегда
 - c. возможно в отдельных случаях

Системное и регрессионное тестирование

1. Каковы особенности системного тестирования?

- a. тестированию подлежит система в целом
- b. тесты оперируют пользовательским или другими внешними интерфейсами
- c. структура проекта тестируется на уровне подсистем
- d. тестирование осуществляется по методу «черного ящика»

2. Какие задачи решаются на уровне системного тестирования?

- a. выявление дефектов в функционировании приложения или в работе с ним
- b. выявление дефектов использования ресурсов
- c. выявление несовместимости с окружением
- d. выявление непредусмотренных сценариев применения
- e. использования непредусмотренных комбинаций данных

3. Какие категории тестов разрабатываются для системного тестирования?

- a. тесты для проверки полноты функциональности
- b. тесты для проверки корректности использования ресурсов
- c. тесты для проверки стрессовых режимов и оценки производительности
- d. тесты для проверки защиты от искаженных данных и некорректных действий
- e. тесты для проверки инсталляции и конфигурации для платформ, предусмотренных спецификацией
- f. тесты для проверки корректности пользовательской документации

4. Каковы особенности регрессионного тестирования?

- a. регрессионное тестирование является подмножеством системного тестирования
- b. выбор между полным и частичным перетестированием и пополнением тестовых наборов
- c. перетестирование предусматривает только контроль частей приложения, связанных с изменениями

5. Какие задачи решает тестировщик, проверяя изменения, внесенные разработчиком в код?

- a. проверка и подтверждение исправления дефекта
- b. обеспечение гарантий, что качество исправленного кода не ухудшилось
- c. перетестирование последствий исправлений на предмет обнаружения индуцированных ошибок
- d. проверка возможности воспроизведения ошибки каким-либо другим способом

6. Какие типы дефектов выявляются при системном или при регрессионном тестировании?

- a. отсутствующая или некорректная функциональность
- b. некорректность проектной документации
- c. непредусмотренные данные или неподдерживаемые сценарии использования
- d. ошибки переносимости на другие платформы
- e. ошибки пользовательской документации
- f. ошибки инсталляции и конфигурирования

7. Можно ли гарантировать безопасность метода регрессионного тестирования при отсутствии информации об изменениях в программе

- a. нет
- b. да

8. Какие методы регрессионного тестирования применяются в условиях высоких требований к качеству программного продукта?

- a. метод повторного прогона всех тестов
- b. случайные методы
- c. безопасные методы
- d. методы минимизации
- e. методы, основанные на покрытии кода

9. Какими свойствами обладает метод повторного прогона всех тестов?

- a. полнота
- b. точность
- c. эффективность
- d. универсальность

10. Можно ли гарантировать безопасность метода регрессионного тестирования в условиях отсутствия тестов, использовавшихся при тестировании предыдущих версий?

- a. тесты, пригодные для повторного использования
- b. тесты, требующие повторного запуска
- c. устаревшие тесты
- d. новые тесты

11. Какие из перечисленных методов тестирования наиболее затратны

- a. системное тестирование с моделируемым окружением
- b. статические методы тестирования
- c. интеграционное тестирование
- d. системное тестирование в реальном окружении и реальном времени
- e. модульное тестирование

12. Какие из перечисленных методов тестирования дают наиболее надежные результаты?

- a. статические методы
- b. модульное тестирование
- c. интеграционное тестирование
- d. системное тестирование с моделируемым окружением

e. системное тестирование в реальном окружении и реальном времени

13. Что влияет на обеспечение качества программного продукта?

- a. тестирование
- b. анализ дизайна
- c. обзоры кода
- d. аудиты процесса разработки ревьюирование проектной документации
- e. ревьюирование проектной документации

14. Какие основные уровни выделены в процессе тестирования?

- a. модульное тестирование
- b. Регрессионное тестирование
- c. интеграционное тестирование
- d. системное тестирование

15. Какие методы регрессионного тестирования применяются в условиях высоких требований к программному продукту?

- a. метод повторного прогона всех тестов
- b. случайные методы
- c. безопасные методы
- d. методы, основанные на покрытии кода
- e. методы минимизации

16. В среднем, метод выборочного регрессионного тестирования...

- a. обнаруживает столько же ошибок, сколько и метод повторного прогона всех тестов
- b. обнаруживает больше ошибок, чем метод повторного прогона всех тестов
- c. обнаруживает меньше ошибок, чем метод повторного прогона всех тестов
- d. может обнаруживать больше или меньше ошибок, чем метод повторного прогона всех тестов

17. На каком этапе регрессионного тестирования удаляются устаревшие тесты?

- a. предсказание целесообразности
- b. идентификация изменений
- c. отбор тестов
- d. выполнение тестов
- e. создание дополнительных тестов
- f. обновление базы данных

18. На каком этапе регрессионного тестирования проводится упорядочение тестов?

- a. предсказание целесообразности
- b. идентификация изменений
- c. отбор тестов
- d. выполнение тестов
- e. создание дополнительных тестов
- f. обновление базы данных

Задания с выбором нескольких вариантов правильных ответов

1. Как реализуются динамические методы построения тестовых путей?

- a) Поиск всех реализуемых путей
- b) Нарращивание начальных отрезков реализованных путей продолжающими их фрагментами, чтобы увеличить покрытие
- c) Построение пути методом удлинения за счет добавления дуг

2. Какие существуют разновидности интеграционного тестирования?

- a) Регрессионное тестирование
- b) Восходящее тестирование
- c) Нисходящее тестирование
- d) Монолитное тестирование

3. Какая информация должна сопровождать действие по исправлению ошибки и перевод дефекта в состояние Resolved?

- a) Краткий комментарий сделанных исправлений
- b) Причину возникновения дефекта
- c) Место исправления дефекта

Тема 1.2.2. Документирование

Вопросы с выбором одного варианта правильного ответа

1. Является ли программа аналогом математической формулы?

- a) Да
- b) Нет
- c) Математические формулы и программы не сводятся друг к другу

2. Какова мощность множества тестов, формально необходимая для тестирования операции в машине с 32-разрядным машинным словом?

- a) 2^{32}
- b) 4^9
- c) 2^{64}

3. Сколько тестов потребуется для проверки программы, реализующей задержку на неопределенное количество тактов?

- a) Один
- b) Неопределенное количество
- c) Зависит от критерия достаточности проверок

4. Зачем нужна спецификация тестирования?

- a) Для формирования команды тестировщиков
- b) Для разработки тестового набора
- c) Для понимания смысла программы

5. Зачем нужен Log-файл?

- a) Для изучения результатов тестирования в режиме on-line
- b) Для фиксации результатов прогона test-suite
- c) Для записи комментариев после прогона тестов

Вопросы с выбором нескольких вариантов правильных ответов

6. Какие подходы используются для обоснования истинности программ?

- a) Использование аналогий
- b) Эксперимент над программой
- c) Доказательство программы

d) Формальный и интерпретационный

7. Отметьте верные утверждения

- a) Тестирование – процесс поиска ошибок
- b) В фазу тестирования входят поиски и исправление ошибок
- c) Отладка – процесс локализации и исправления ошибок

8. Какие существуют методы анализа и локализации ошибки?

- a) Выполнение программы в уме
- b) Пошаговое выполнение
- c) Метод контрольных точек и анализа трасс

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Вопросы с выбором одного варианта правильного ответа

1. Какие существуют особенности управляющего графа программы в случае объектно-ориентированного программирования (ООП)?

- a. она становится неприменимой
- b. она требует адаптации по обработке сообщений
- c. она требует описания поведения программы
- d. она требует описания не только структуры, но и поведения программы

2. Какие существуют особенности интеграционного тестирования для ООП?

- a. тестирование дерева классов
- b. тестирование последовательностей прямых вызовов методов с помощью Р-путей
- c. тестирование последовательностей обработки сообщений с помощью ММ-путей
- d. тестирование исключительно цепочек вызовов процедур программ

3. Какие этапы включает методика ООП при тестировании программного комплекса?

- a. тестирование методов каждого класса программного комплекса
- b. тестирование отношений между классами с помощью тестов на основе Р-путей или ММ-путей
- c. тестирование взаимодействия модулей по всей иерархии комплекса

4. Какие этапы методики тестирования используются в ООП?

- a. тестирование методов каждого класса программы
- b. тестирование методов класса, входящих в его контекст
- c. тестирование дерева классов программного проекта, включающего оттестированный класс

5. Какие этапы методов тестирования класса используются в ООП?

- a. тестирование класса как модуля по выбранному критерию
- b. тестирование класса как иерархической структуры
- c. тестирование классов, входящих в модель проекта

6. Что такое управляющий граф программы (УГП)?

- a. множество операторов программы
- b. граф, вершины которого кодируют операторы программы, а дуги - управления (порядок исполнения) операторов
- c. множество операторов управления

7. Что такое путь в УГП?

- a. последовательность вершин и дуг управляющего графа программы с фиксированными начальной и конечной вершиной
- b. последовательность ветвей управляющего графа программы с фиксированными начальной вершиной первой ветви и конечной вершиной последней ветви пути
- c. множество связанных дуг управляющего графа программы

Вопрос 8. Что такое ветвь управляющего графа программы?

- a) последовательность вершин и дуг управляющего графа программы с фиксированными начальной и конечной вершиной, которые кодируют либо условные операторы, либо первый и последний операторы управляющего графа программы соответственно
- b. часть пути, в котором все внутренние вершины кодируют линейные операторы
- c. начальная и конечная вершина пути

9. Можно ли гарантировать остановку программы на любом тесте?

- a. в общем случае нет
- b. возможно в частных случаях
- c. задача в общей постановке алгоритмически неразрешима

10. Какие задачи у модульного тестирования?

- a. выявление локальных ошибок реализации алгоритмов модулей
- b. выявление ошибок при вызове модулей
- c. выявление ошибок взаимодействия модуля с окружением

11. На основе каких принципов строятся тесты для модульного тестирования?

- a. анализ потоков управления модуля
- b. анализ потоков данных модуля
- c. анализ покрытия в соответствии с заданными критериями C0, C1, C2

12. Каковы фазы процесса построения тестовых путей?

- a. построение управляющего графа программы
- b. выбор тестовых путей
- c. генерация тестов, соответствующих выбранным тестовым путям

13. Какие существуют методы построения тестовых путей?

- a. статические
- b. динамические
- c. методы реализуемых путей

14. Как реализуются динамические методы построения тестовых путей?

- a. наращивание начальных отрезков реализованных путей продолжающими их фрагментами, чтобы увеличить покрытие
- b. построение пути методом удлинения за счет добавления дуг
- c. поиск всех реализуемых путей

15. Какие существуют разновидности интеграционного тестирования?

- a. монолитное тестирование
- b. нисходящее тестирование
- c. восходящее тестирование
- d. Регрессионное тестирование

16. Каковы особенности нисходящего тестирования?

- a. необходимость разработки заглушек

- b. необходимость разработки среды управления очередностью вызовов модулей
- c. параллельная разработка эффективных модулей

17. Каковы особенности восходящего тестирования?

- a. минимизация разработки заглушек
- b. запаздывание в проверке функциональности реализуемого приложения
- c. необходимость разработки среды управления очередностью вызовов модулей

18. В чем заключаются особенности интеграционного тестирования для процедурного программирования?

- a. тестирование программных комплексов, заданных в виде иерархических структур модулей
- b. использование диаграмм потока управления в качестве модели тестируемого комплекса
- c. контроль соответствия спецификациям параметров модулей и межмодульных связей
- d. контроль наследования

19. При создании очередной версии программы была добавлена функция А, функция D была удалена, функция С – изменена. К какой группе относится тест, покрывающий только функцию А?

- a. тесты, пригодные для повторного использования
- b. тесты, требующие повторного запуска
- c. устаревшие тесты
- d. новые тесты

20. Возможно ли тестирование программы на всех допустимых значениях параметров?

- a. никогда
- b. да, всегда
- c. возможно в отдельных случаях

Системное и регрессионное тестирование

1. Каковы особенности системного тестирования?

- a. тестированию подлежит система в целом
- b. тесты оперируют пользовательским или другими внешними интерфейсами
- c. структура проекта тестируется на уровне подсистем
- d. тестирование осуществляется по методу «черного ящика»

2. Какие задачи решаются на уровне системного тестирования?

- a. выявление дефектов в функционировании приложения или в работе с ним
- b. выявление дефектов использования ресурсов
- c. выявление несовместимости с окружением
- d. выявление непредусмотренных сценариев применения
- e. использования непредусмотренных комбинаций данных

3. Какие категории тестов разрабатываются для системного тестирования?

- a. тесты для проверки полноты функциональности
- b. тесты для проверки корректности использования ресурсов
- c. тесты для проверки стрессовых режимов и оценки производительности
- d. тесты для проверки защиты от искаженных данных и некорректных действий
- e. тесты для проверки инсталляции и конфигурации для платформ, предусмотренных спецификацией
- f. тесты для проверки корректности пользовательской документации

4. Каковы особенности регрессионного тестирования?

- a. регрессионное тестирование является подмножеством системного тестирования

- b. выбор между полным и частичным перетестированием и пополнением тестовых наборов
- c. перетестирование предусматривает только контроль частей приложения, связанных с изменениями

5. Какие задачи решает тестировщик, проверяя изменения, внесенные разработчиком в код?

- a. проверка и подтверждение исправления дефекта
- b. обеспечение гарантий, что качество исправленного кода не ухудшилось
- c. перетестирование последствий исправлений на предмет обнаружения индуцированных ошибок
- d. проверка возможности воспроизведения ошибки каким-либо другим способом

6. Какие типы дефектов выявляются при системном или при регрессионном тестировании?

- a. отсутствующая или некорректная функциональность
- b. некорректность проектной документации
- c. непредусмотренные данные или неподдерживаемые сценарии использования
- d. ошибки переносимости на другие платформы
- e. ошибки пользовательской документации
- f. ошибки инсталляции и конфигурирования

7. Можно ли гарантировать безопасность метода регрессионного тестирования при отсутствии информации об изменениях в программе

- a. нет
- b. да

8. Какие методы регрессионного тестирования применяются в условиях высоких требований к качеству программного продукта?

- a. метод повторного прогона всех тестов
- b. случайные методы
- c. безопасные методы
- d. методы минимизации
- e. методы, основанные на покрытии кода

9. Какими свойствами обладает метод повторного прогона всех тестов?

- a. полнота
- b. точность
- c. эффективность
- d. универсальность

10. Можно ли гарантировать безопасность метода регрессионного тестирования в условиях отсутствия тестов, использовавшихся при тестировании предыдущих версий?

- a. тесты, пригодные для повторного использования
- b. тесты, требующие повторного запуска
- c. устаревшие тесты
- d. новые тесты

11. Какие из перечисленных методов тестирования наиболее затратны

- a. системное тестирование с моделируемым окружением
- b. статические методы тестирования
- c. интеграционное тестирование
- d. системное тестирование в реальном окружении и реальном времени
- e. модульное тестирование

12. Какие из перечисленных методов тестирования дают наиболее надежные результаты?

- a. статические методы
- b. модульное тестирование
- c. интеграционное тестирование
- d. системное тестирование с моделируемым окружением
- e. системное тестирование в реальном окружении и реальном времени

13. Что влияет на обеспечение качества программного продукта?

- a. тестирование
- b. анализ дизайна
- c. обзоры кода
- d. аудиты процесса разработки ревьюирование проектной документации
- e. ревьюирование проектной документации

14. Какие основные уровни выделены в процессе тестирования?

- a. модульное тестирование
- b. Регрессионное тестирование
- c. интеграционное тестирование
- d. системное тестирование

15. Какие методы регрессионного тестирования применяются в условиях высоких требований к программному продукту?

- a. метод повторного прогона всех тестов
- b. случайные методы
- c. безопасные методы
- d. методы, основанные на покрытии кода
- e. методы минимизации

16. В среднем, метод выборочного регрессионного тестирования...

- a. обнаруживает столько же ошибок, сколько и метод повторного прогона всех тестов
- b. обнаруживает больше ошибок, чем метод повторного прогона всех тестов
- c. обнаруживает меньше ошибок, чем метод повторного прогона всех тестов
- d. может обнаруживать больше или меньше ошибок, чем метод повторного прогона всех тестов

17. На каком этапе регрессионного тестирования удаляются устаревшие тесты?

- a. предсказание целесообразности
- b. идентификация изменений
- c. отбор тестов
- d. выполнение тестов
- e. создание дополнительных тестов
- f. обновление базы данных

18. На каком этапе регрессионного тестирования проводится упорядочение тестов?

- a. предсказание целесообразности
- b. идентификация изменений
- c. отбор тестов
- d. выполнение тестов
- e. создание дополнительных тестов
- f. обновление базы данных

19. Является ли программа аналогом математической формулы?

- a) Да
- b) Нет
- c) Математические формулы и программы не сводятся друг к другу

20. Какова мощность множества тестов, формально необходимая для тестирования операции в машине с 32-разрядным машинным словом?

- a) 2^{32}
- b) 4^9
- c) 2^{64}

21. Сколько тестов потребуется для проверки программы, реализующей задержку на неопределенное количество тактов?

- a) Один
- b) Неопределенное количество
- c) Зависит от критерия достаточности проверок

22. Зачем нужна спецификация тестирования?

- a) Для формирования команды тестировщиков
- b) Для разработки тестового набора
- c) Для понимания смысла программы

23. Зачем нужен Log-файл?

- a) Для изучения результатов тестирования в режиме on-line
- b) Для фиксации результатов прогона test-suite
- c) Для записи комментариев после прогона тестов

Вопросы с выбором нескольких вариантов правильных ответов

1. Как реализуются динамические методы построения тестовых путей?

- a) Поиск всех реализуемых путей
- b) Нарращивание начальных отрезков реализованных путей продолжающими их фрагментами, чтобы увеличить покрытие
- c) Построение пути методом удлинения за счет добавления дуг

2. Какие существуют разновидности интеграционного тестирования?

- a) Регрессионное тестирование
- b) Восходящее тестирование
- c) Нисходящее тестирование
- d) Монолитное тестирование

3. Какая информация должна сопровождать действие по исправлению ошибки и перевод дефекта в состояние Resolved?

- a) Краткий комментарий сделанных исправлений
- b) Причину возникновения дефекта
- c) Место исправления дефекта

4. Какие подходы используются для обоснования истинности программ?

- a) Использование аналогий
- b) Эксперимент над программой
- c) Доказательство программы
- d) Формальный и интерпретационный

5. Отметьте верные утверждения

- a) Тестирование – процесс поиска ошибок
- b) В фазу тестирования входят поиски и исправление ошибок
- c) Отладка – процесс локализации и исправления ошибок

6. Какие существуют методы анализа и локализации ошибки?

- a) Выполнение программы в уме
- b) Пошаговое выполнение
- c) Метод контрольных точек и анализа трасс

3.4. МДК.01.03 Разработка мобильных приложений

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ ЗНАНИЙ

Вопросы с выбором одного варианта правильного ответа

1. Что такое мобильное приложение?

- a) Программа, которая устанавливается на мобильное устройство и выполняет определенные функции.
- b) Веб-приложение, которое работает только в браузере.
- c) Приложение, которое работает только на настольных компьютерах.

2. Какие языки программирования используются для разработки мобильных приложений?

- a) JavaScript, Python, Java, Swift.
- b) JavaScript, Python, C#, Swift.
- c) JavaScript, Python, Java, C#.

3. Что такое React Native?

- a) Фреймворк для разработки мобильных приложений на JavaScript.
- b) Библиотека для создания веб-приложений.
- c) Язык программирования для создания игр.

4. Какие преимущества имеет разработка мобильных приложений на JavaScript?

- b) Высокая скорость разработки, возможность быстрого прототипирования.
- c) Легкость интеграции с веб-сервисами и базами данных.
- a) Кроссплатформенность, возможность использования одного кода для разных платформ.

5. Какие инструменты используются для разработки мобильных приложений на JavaScript?

- a) React Native, Ionic, Cordova.
- b) React Native, Angular, Vue.js.
- c) React Native, Node.js, Express.js.

6. Какие функции можно реализовать в мобильном приложении на JavaScript?

- a) Работа с камерой, GPS, уведомлениями.
- b) Работа с базами данных, веб-сервисами.
- c) Работа с графикой, анимацией.

7. Какие преимущества имеет React Native перед нативными мобильными приложениями?

- a) Более быстрая разработка, возможность использования одного кода для разных платформ.
- b) Более низкая стоимость разработки, возможность использования одного кода для разных платформ.
- c) Более легкая интеграция с веб-сервисами и базами данных.

8. Какие платформы поддерживает React Native?

- a) iOS и Android.
- b) iOS, Android, Windows Phone.
- c) iOS, Android, BlackBerry.

9. Какие преимущества имеет разработка мобильных приложений на JavaScript перед разработкой нативных приложений?

- a) Более быстрая разработка, возможность использования одного кода для разных платформ.
- b) Более низкая стоимость разработки, возможность использования одного кода для разных платформ.
- c) Более легкая интеграция с веб-сервисами и базами данных.

10. Какие инструменты используются для тестирования мобильных приложений на JavaScript?

- a) Jest, Enzyme, Detox.
- b) Mocha, Chai, Sinon.
- c) Karma, Jasmine, Protractor.

11. Какие инструменты используются для сборки и развертывания мобильных приложений на JavaScript?

- a) npm, yarn, webpack.
- b) npm, yarn, gulp.
- c) npm, yarn, grunt.

12. Какие преимущества имеет разработка мобильных приложений на JavaScript перед разработкой нативных приложений?

- a) Более быстрая разработка, возможность использования одного кода для разных платформ.
- b) Более низкая стоимость разработки, возможность использования одного кода для разных платформ.
- c) Более легкая интеграция с веб-сервисами и базами данных.

13. Какие инструменты используются для создания пользовательского интерфейса в мобильных приложениях на JavaScript?

- a) React Native, Ionic, Cordova.
- b) React Native, Angular, Vue.js.
- c) React Native, Node.js, Express.js.

14. Что такое JavaScript?

- a) Язык программирования, используемый для создания веб-страниц.
- b) Язык программирования, используемый для создания мобильных приложений.
- c) Язык программирования, используемый для создания игр.

15. Какие основные принципы лежат в основе JavaScript?

- a) Объектно-ориентированное программирование.
- b) Функциональное программирование.
- c) Прототипное программирование.

16. Какие основные конструкции языка JavaScript?

- a) Переменные, функции, операторы, циклы.
- b) Классы, методы, свойства, наследование.
- c) Объекты, массивы, прототипы, замыкания.

17. Какие возможности предоставляет JavaScript для работы с веб-страницами?

- a) Изменение стилей элементов на странице.
- b) Обработка событий на странице.
- c) Создание и манипуляция элементами на странице.

18. Какие библиотеки и фреймворки могут использоваться вместе с JavaScript для разработки веб-приложений?

- a) React, Angular, Vue.js.

- b) jQuery, MooTools, Prototype.
- c) Node.js, Express.js, Socket.io.

Вопросы с выбором нескольких вариантов правильных ответов

1. Какие методы отражают жизненный цикл Activity:

- a. onCreate() ()
- b. onRestart() ()
- c. onResume() ()
- d. onRunning() ()
- e. onClose() ()

Задачи, модельные ситуации или проблемы, ориентированные на профессиональную деятельность

Задание 1. Скачать Android SDK Eclipse (Eclipse ADT Bundle) <http://developer.android.com/sdk/index.html>, а также последнюю версию Java (JDK) <http://www.oracle.com/technetwork/java/index-jsp-138363.html> (или просто набрать в yandex "JDK" и пройти по ссылке на oracle.com). Установить всё это. Создать новый проект, зайти в Android SDK скачать какую-нибудь версию Android SDK (например, lollipop - 5.0.1),

Задание 2. Создать приложение, содержащее анимированные интерфейсные элементы (например, увеличивающиеся при клике на них кнопки, вращающиеся TextView и т.д.).

Задание 3. Создать приложение, отображающее после запуска карты Google или какие-нибудь другие карты.

Задание 4. Создать приложение, использующее опциональное меню (меню настроек) и контекстное меню для какого-нибудь интерфейсного элемента. Естественно, выбор пунктов меню должен что-то менять в интерфейсных элементах или их отображении! Например, очистить поле ввода через контекстное меню, или отобразить невидимые интерфейсные элементы через установку галок в опциональном меню.

Задание 5. Создать приложение, отображающее после некоторых действий (нажатия на кнопку, например, или проверки корректности ввода текста в EditText) диалоговое окно, свидетельствующее об ошибке/информирующее/предупреждающее пользователя о чём-то.

Задание 6. Создать приложение, помещающее по нажатию на кнопку какое-то сообщение со звуком в панель уведомлений/статус-панель на эмуляторе.

Задание 7. Создать приложение, по нажатию кнопки в котором проигрывается какой-то звук.

Задание 7. Создать приложение, при запуске которого проигрывается какое-то видео.

Дать развернутый ответ на вопрос

1. Каковы основные этапы разработки мобильного приложения?

2. **Какие основные компоненты включает в себя мобильное приложение?**
3. **Какие технологии обычно используются при разработке мобильных приложений?**
4. **Что такое кроссплатформенная разработка мобильных приложений и каковы ее преимущества?**
5. **Какие виды монетизации мобильных приложений существуют?**

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Вопросы с выбором одного варианта правильного ответа

1. **Набор средств программирования, который содержит инструменты, необходимые для создания, компиляции и сборки мобильного приложения называется:**
 - а) Android SDK
 - б) JDK
 - в) плагин ADT
 - г) Android NDK
2. **С какой целью был создан Open Handset Alliance?**
 - а) писать историю развития ОС Android
 - б) продавать смартфоны под управлением Android
 - в) рекламировать смартфоны под управлением Android
 - г) разрабатывать открытые стандарты для мобильных устройств
3. **С какой целью инструмент Intel* Graphics Performance Analyzers (Intel* GPA) SystemAnalyzer используется в среде разработки Intel* Beacon Mountain?**
 - а) позволить разработчикам оптимизировать загрузку системы при использовании процедур OpenGL
 - б) для ускорения работы эмулятора в среде разработки
 - в) для оптимизированной обработки данных и изображений
 - г) позволить разработчикам эффективно распараллелить C++ мобильные приложения
4. **Библиотеки, реализованные на базе PacketVideo OpenCORE:**
 - а) Media Framework

- б) SQLite
- в) FreeType
- г) 3D библиотеки

5. Какой движок баз данных используется в ОС Android?

- а) InnoDB
- б) DBM
- в) MyISAM
- г) SQLite

6. С какой целью инструмент Intel* Integrated Performance Primitives (Intel* IPP) используется в среде разработки Intel* Beacon Mountain?

- а) для оптимизированной обработки данных и изображений
- б) позволить разработчикам оптимизировать загрузенность системы при использовании процедур OpenGL
- в) для ускорения работы эмулятора в среде разработки
- г) позволить разработчикам эффективно распараллелить C мобильные приложения

7. Intel XDK поддерживает разработку под:

- а) JavaFX Mobile
- б) Apple iOS, BlackBerry OS
- в) MtkOS, Symbian OS, Microsoft Windows 8
- г) Android, Apple iOS, Microsoft Windows 8, Tizen

8. Каждый приемник широковещательных сообщений является наследником класса...

- а) ViewReceiver
- б) IntentReceiver
- в) ContentProvider
- г) BroadcastReceiver

9. Какой класс является основным строительным блоком для компонентов пользовательского интерфейса (UI), определяет прямоугольную область экрана и отвечает за прорисовку и обработку событий?

- а) GUI
- б) View
- в) UIComponent
- г) Widget

10. Какой слушатель используется для отслеживания события касания экрана устройства?

- а) OnPressListener
- б) OnTouchListener
- в) nClickListener
- г) OnInputListener

11. В какой папке необходимо разместить XML файлы, которые определяют все меню приложения?

- а) res/value
- б) res/items

- в) res/layout
- г) res/menu

12. Фоновые приложения ...

- а) после настройки не предполагают взаимодействия с пользователем, большую часть времени находятся и работают в скрытом состоянии
- б) выполняют свои функции и когда видимы на экране, и когда скрыты другими приложениями
- в) небольшие приложения, отображаемые в виде графического объекта на рабочем столе
- г) большую часть времени работают в фоновом режиме, однако допускают взаимодействие с пользователем и после настройки

13. Какой компонент управляет распределенным множеством данных приложения?

- а) сервис (Service)
- б) активность (Activity)
- в) приемник широковещательных сообщений (BroadcastReceiver)
- г) контент-провайдер (Content Provider)

14. Какой язык разметки используется для описания иерархии компонентов графического пользовательского интерфейса Android-приложения?

- а) html
- б) xml
- в) gml
- г) xhtml

15. Выберите верную последовательность действий, необходимых для создания в приложении контент-провайдера.

- а) Создание класса наследника от класса ContentProvider; Определение способа организации данных; Заполнение контент-провайдера данными
- б) Проектирование способа хранения данных; Определение способа организации данных;
- в) Создание класса наследника от класса ContentProvider; Заполнение контент-провайдера данными; Определение способа работы с данными
- г) Проектирование способа хранения данных; Создание класса наследника от класса ContentProvider; Определение строки авторизации провайдера, URI для его строк и имен столбцов

16. Какие элементы управления применяются для действий по настройке?

- а) командные элементы управления
- б) элементы выбора
- в) элементы ввода
- г) элементы отображения

17. Примерами комбо-элементов не являются:

- а) комбо-список
- б) все выше перечисленное
- в) комбо-кнопка
- г) комбо-поле

18. Более крупные элементы:

- а) привлекают больше внимания
- б) все варианты ответа верны
- в) размер не влияет на уровень внимания
- г) привлекают меньше внимания

19. К традиционным типографическим инструментам не относят

- а) масштаб
- б) цвет
- в) разреженность
- г) выравнивание по сетке

20. К элементам ввода относят:

- а) ограничивающие элементы ввода
- б) ползунки
- в) счетчики

21. Выделяют следующие категории плотности экрана для Android-устройств:

- а) LDPI, XHDPI, XXHDPI, и XXXHDPI
- б) правильный вариант ответа отсутствует
- в) LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI
- г) LDPI, MDPI, HDPI

22. Основные вкладки (FixedTabs) удобны при отображении

- а) от четырех вкладок
- б) двух вкладок
- в) трех и более вкладок
- г) трех и менее вкладок

23. Диалоговое окно, содержащее линейку процесса выполнения какого-то действия – это

- а) DatePickerDialog
- б) AlertDialog
- в) ProgressDialog
- г) DialogFragment

24. Уведомления стоит использовать, когда

- а) сообщение не требует ответа пользователя, но важно для продолжения его работы
- б) сообщение является важным и требует немедленного прочтения и ответа
- в) сообщение является важным, однако требует немедленного прочтения, но не ответа
- г) сообщение является важным, однако не требует немедленного прочтения и ответа

25. Какой метод запускает новую активность?

- а) startActivity ()
- б) beginActivity

- д) `startActivity()`
- в) `intentActivity()`
- г) `newActivity()`

Вопросы с выбором нескольких вариантов правильных ответов

1. Полный иерархический список обязательных файлов и папок проекта можно увидеть на вкладке ...

- а) Package Explorer
- б) Internet Explorer
- в) Navigator
- г) Project Explorer

2. Выберите верные утверждения относительно объекта-намерения (Intent).

- а) представляет собой структуру данных, содержащую описание операции, которая должна быть выполнена, и обычно используется для запуска активности или сервиса
- б) используется для передачи сообщений пользователю
- в) используется для получения инструкций от пользователя
- г) используются для передачи сообщений между основными компонентами приложений

3. Расположение элементов мобильного приложения:

- а) полезно для передачи иерархии
- б) влияет на удобство использования
- в) полезно для создания пространственных отношений между объектами на экране и объектами реального мира

4. Дизайн или проектирование интерфейса для графических дизайнеров:

- а) прозрачность и понятность информации
- б) тон, стиль, композиция, которые являются атрибутами бренда
- в) передача информации о поведении посредством ожидаемого назначения

Задачи, модельные ситуации или проблемы, ориентированные на профессиональную деятельность

Задание 1. Скачать Android SDK Eclipse (Eclipse ADT Bundle)

<http://developer.android.com/sdk/index.html>, а также последнюю версию Java (JDK) <http://www.oracle.com/technetwork/java/index-jsp-138363.html> (или просто набрать в yandex "JDK" и пройти по ссылке на oracle.com). Установить всё это. Создать новый проект, зайти в Android SDK и скачать какую-нибудь версию Android SDK (например, lollipop - 5.0.1),

Задание 2. Создать приложение, содержащее анимированные интерфейсные элементы (например, увеличивающиеся при клике на них кнопки, вращающиеся TextView и т.д.).

Задание 3. Создать приложение, отображающее после запуска карты Google или какие-нибудь другие карты.

Задание 4. Создать приложение, использующее опциональное меню (меню настроек) и контекстное меню для какого-нибудь интерфейсного элемента. Естественно, выбор пунктов меню должен что-то менять в интерфейсных элементах или их отображении!

Например, очистить поле ввода через контекстное меню, или отобразить невидимые интерфейсные элементы через установку галок в опциональном меню.

Задание 5. Создать приложение, отображающее после некоторых действий (нажатия на кнопку, например, или проверки корректности ввода текста в EditText) диалоговое окно, свидетельствующее об ошибке/информирующее/предупреждающее пользователя о чём-то.

Задание 6. Создать приложение, помещающее по нажатию на кнопку какое-то сообщение со звуком в панель уведомлений/статус-панель па эмуляторе.

Задание 7. Создать приложение, по нажатию кнопки в котором проигрывается какой-то звук.

Задание 7. Создать приложение, при запуске которого проигрывается какое-то видео.

Дать развернутый ответ на вопрос

- 1. Каковы основные этапы разработки мобильного приложения?**
- 2. Какие основные компоненты включает в себя мобильное приложение?**
- 3. Какие технологии обычно используются при разработке мобильных приложений?**
- 4. Что такое кроссплатформенная разработка мобильных приложений и каковы ее преимущества?**
- 5. Какие виды монетизации мобильных приложений существуют?**

3.5. МДК 01.04 Системное программирование

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

- 1 Установите порядок, в котором осуществляется перевод произвольного двоичного числа в системе счисления с основанием $q=2n$. Укажите порядок следования всех 3 вариантов ответа:** 1 рассмотреть каждую группу как n -разрядное двоичное число и записать ее соответствующей цифрой в системе счисления с основанием $q = 2n$;
2 двоичное число разбить слева и справа (целую и дробную части) на группы по цифр в каждой;
3 если в последних правой и левой группах окажется меньше разрядов, то их надо дополнить справа и слева нулями до нужного числа разрядов
- a. 3 1 2
b. 2 1 3
c. 1 2 3
d. 3 2 1
- 2 По сколько цифр в группе нужно разбивать двоичное число при переводе его в четверичную систему счисления?**
- a. 6
b. 2
c. 3
d. 4
- 3 Переведите двоичное число 100011111011 в шестнадцатеричную систему счисления:**
- a. 5fa
b. 8fb
c. 8fc
d. 8fa
- 4 Чему будет равен результат сложение двоичных чисел 1001 и 1000**
- a. 10001
b. 11001
c. 10011
d. 10101
- 5 Выполните деление в двоичной системе счисления $1110:10=$**
- a. 101
b. 011
c. 110
d. 111
- 6 Переведите двоичное число 1010101 в восьмеричную систему счисления**
- a. 125
b. 124
c. 126
d. 123
- 7 Установите порядок, в котором осуществляется перевод целого числа в системе счисления с основанием $q=2n$. Укажите порядок следования всех 3 вариантов ответа:** 1 данное двоичное число разбить справа налево на группы по цифр в каждой;
2 рассмотреть каждую группу как n -разрядное двоичное число и записать ее соответствующей цифрой в системе счисления с основанием $q = 2n$;
3 если в последней левой группе окажется меньше разрядов, то ее надо дополнить слева нулями до нужного числа разрядов.
- 1 3 2
a. 1 2 3
c. 3 2 1
d. 2 1 3
- 8 Переведите шестнадцатеричное число FACC в двоичную систему счисления**

- a. 1111001011011001
- b. 1111101011001100
- c. 1011010111111100
- d. 1101100111011010

9 Выполните вычисление в двоичной системе счисления $110101-101=$

- a. 110100
- b. 110010
- c. 110000
- d. 110001

10 В саду 100 фруктовых деревьев - 14 яблонь и 42 груши. В какой системе счисления посчитаны деревья?

- a. В шестеричной
- b. В шестнадцатеричной
- c. В двоичной
- d. В восьмеричной

11 Умножьте в двоичной системе счисления $1111*11=$

- a. 1111111
- b. 101111
- c. 1010101
- d. 101101

13 Выполните умножение в системе счисления $10101*101=$

- a. 110101011
- b. 1101001
- c. 111001001
- d. 1010101

15 В какой системе счисления будет верным равенство $7 \cdot 8=16?$

- a. 12
- b. 6
- c. 9
- d. 16

16 FFFF \cdot 1=

- a. 1FFF
- b. 10000
- c. FFFE
- d. FFF1

17 Выполните вычитание в двоичной системе счисления $1111-101$

- a. 1010
- b. 1110
- c. 1100
- d. 1001

18 Результат деления в двоичной системе счисления числа 110 на 11 равен

- a. 11
- b. 10
- c. 01
- d. 101

19 Перевести смешанное двоичное $111100000,101$ число в восьмеричную систему счисления

- a. 610,4
- b. 740,5
- c. 170,5
- d. 1E0,A

20 Переведите восьмеричное число 0,25 в двоичную систему счисления

- a. 0,010101
- b. 0,101010
- c. 0,110111
- d. 0,101011

21 Алгоритм-это:

- a. Указание на выполнение действий,
- b. Система правил, описывающая последовательность действий, которые необходимо выполнить для решения задачи,
- c. Процесс выполнения вычислений, приводящих к решению задачи

22 Свойство алгоритма – дискретность, выражает, что:

- a. Команды должны следовать последовательно друг за другом,
- b. Каждая команда должна быть описана в расчете на конкретного исполнителя,
- c. Разбиение алгоритма на конечное число команд

23 Формальное исполнение алгоритма – это:

- a. Исполнение алгоритма конкретным исполнителем с полной записью его рассуждений,
- b. Разбиение алгоритма на конкретное число команд и пошаговое их исполнение,
- c. Исполнение алгоритма не требует рассуждений, а осуществляется исполнителем автоматически

- d. Исполнение алгоритма осуществляется исполнителем на уровне его знаний

24 Самое важное свойство алгоритма:

- a. Визуальность,
- b. Массовость,
- c. Дискретность,
- d. Аудиальность,

25 Какой алгоритм называется линейным:

- a. Выполнение операций зависит от условия,
- b. Операции выполняются друг за другом,
- c. Одни и те же операции выполняются многократно
- d. Присутствие всех возможных операций в одном алгоритме

26 Графические задание алгоритма – это:

- a. Способ представления алгоритма с помощью геометрических фигур,
- b. Представление алгоритма в форме таблиц и расчетных формул,
- c. Система обозначений и правил для единообразной и точной записи алгоритмов и их исполнения

27 В расчете на кого должен строиться алгоритм:

- a. В расчете на ЭВМ,
- b. В расчете на умственные способности товарища,
- c. В расчете на конкретного исполнителя

28 Какое из перечисленных свойств относится к свойствам алгоритма:

- a. Визуальность,
- b. Совокупность,
- c. Аудиальность,
- d. Понятность

29 Псевдокоманда DW означает

- a. определить байт;
- b. определить слово (2 байта);
- c. определить двойное слово (4 байта);
- d. определить 6 байт
- e. определить учетверенное слово (8 байт);

30 Псевдокоманда DW означает

- a. определить байт;
- b. определить слово (2 байта);
- c. определить двойное слово (4 байта);
- d. определить 6 байт
- e. определить учетверенное слово (8 байт);

4. КРИТЕРИИ ОЦЕНКИ

Для обучающихся по образовательным программам среднего профессионального образования применяется пятибалльная шкала знаний, умений, практического опыта.

Шкалы оценивания		Критерии оценивания
пятибалльная	зачет	
«Отлично» - 5 баллов		<p>Показывает высокий уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует глубокое и прочное освоение материала; – исчерпывающе, четко, последовательно, грамотно и логически стройно излагает теоретический материал; – правильно формирует определения; – демонстрирует умения самостоятельной работы с нормативно-правовой литературой; – умеет делать выводы по излагаемому материалу.
«Хорошо» - 4 балла		<p>Показывает достаточный уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует достаточно полное знание материала, основных теоретических положений; – достаточно последовательно, грамотно логически стройно излагает материал; – демонстрирует умения ориентироваться в нормативно-правовой литературе; – умеет делать достаточно обоснованные выводы по излагаемому материалу.
«Удовлетворительно» - 3 балла	Зачтено	<p>Показывает пороговый уровень сформированности компетенций, т.е.:</p> <ul style="list-style-type: none"> – демонстрирует общее знание изучаемого материала; – испытывает затруднения при ответах на дополнительные вопросы; – знает основную рекомендуемую литературу; – умеет строить ответ в соответствии со структурой излагаемого материала.
«Неудовлетворительно» - 2 балла	Не зачтено	<p>Ставится в случае:</p> <ul style="list-style-type: none"> – незнания значительной части программного материала; – не владения понятийным аппаратом курса; – допущения существенных ошибок при изложении учебного материала; – неумения строить ответ в соответствии со структурой излагаемого вопроса; – неумения делать выводы по излагаемому материалу.

Критерии оценки тестовых заданий

Процент выполненных тестовых заданий	Оценка
до 50%	неудовлетворительно
50-69%	удовлетворительно
70-84%	хорошо
85-100%	отлично

Критерии оценки тестовых заданий, заданий на дополнение, с развернутым ответом и на установление правильной последовательности

Верный ответ - 2 балла.

Неверный ответ или его отсутствие - 0 баллов.

Критерии оценки заданий на сопоставление

Верный ответ - 2 балла

1 ошибка - 1 балл

более 1-й ошибки или ответ отсутствует - 0 баллов.