

МИНОБРНАУКИ РОССИИ  
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ЭКОНОМИКИ И СЕРВИСА

**РАБОЧАЯ ПРОГРАММА  
ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

*ПМ.03 Участие в интеграции программных модулей*

программы подготовки специалистов среднего звена  
*09.02.03 Программирование в компьютерных системах*

на базе среднего общего образования

Форма обучения: *очная*


Рабочая программа профессионального модуля *ПМ.03 «Участие в интеграции программных модулей»* разработана в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.03 Программирование в компьютерных системах, утвержденного приказом Минобрнауки России от «28» июля 2014 г. № 804, примерной образовательной программой.

Разработчик(и):

Коломийцев А.К., преподаватель ОСПО филиала ФГБОУ ВО «ВГУЭС» в г. Находке

Рассмотрено и одобрено на заседании межпредметной цикловой методической комиссии

Протокол № 9 от «30» марта 2020 г.

Председатель ЦМК  Н.П. Фадеева  
подпись

## **СОДЕРЖАНИЕ**

- 1 ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**
- 2 СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**
- 3 УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**
- 4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ**

## 1 ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

### 1.1 Место профессионального модуля в структуре основной образовательной программы

Профессиональный модуль ПМ.03 «Участие в интеграции программных модулей» является частью профессионального учебного цикла основной образовательной программы (далее ООП) в соответствии с ФГОС СПО по специальности 09.02.03 Программирование в компьютерных системах.

### 1.2 Цель и планируемые результаты освоения профессионального модуля

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

Код компетенции	Наименование общих компетенций	Результаты освоения
ОК 1	Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.	<b>практический опыт:</b> - выполнять профессиональные задачи при выполнении выпускной квалификационной работы; - проявлять творческую инициативу, демонстрировать профессиональную подготовку <b>умения:</b> - овладеть первичными профессиональными навыками и умениями; - планировать будущую профессиональную деятельность <b>знания:</b> - иметь представление о будущей профессии; - ориентироваться в маршруте студента по специальности; - называть основные виды работ, выполняемые при работе по специальности
ОК 2	Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.	<b>практический опыт:</b> - планирования деятельности, применяя технологию с учетом изменения параметров объекта; - выбирать типовой способ (технологию) решения задачи в соответствии с заданными условиями, имеющимися ресурсами, критериями качества и эффективности <b>умения:</b> - планировать деятельность по решению задачи в рамках заданных (известных) технологий, в том числе выделяя отдельные составляющие технологии;

		<ul style="list-style-type: none"> <li>- анализировать потребности в ресурсах и планировать ресурсы в соответствии с заданным способом решения задачи</li> </ul> <p><b>знания:</b></p> <ul style="list-style-type: none"> <li>- методы и способы выполнения профессиональных задач;</li> <li>- называть ресурсы для решения поставленной задачи в соответствии с заданным способом деятельности</li> </ul>
ОК 3	Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	<p><b>практический опыт:</b></p> <ul style="list-style-type: none"> <li>- проводить анализ причин существования проблемы;</li> <li>- предлагать способ коррекции деятельности на основе результатов текущего контроля и результатов оценки продукта деятельности;</li> <li>- определять показатели результативности деятельности в соответствии с поставленной профессиональной задачей;</li> <li>- задавать критерии для определения способа разрешения проблемы;</li> <li>- прогнозировать последствия принятых решений;</li> <li>- называть риски на основе самостоятельно проведенного анализа ситуации;</li> <li>- предлагать способы предотвращения и нейтрализации рисков</li> </ul> <p><b>умения:</b></p> <ul style="list-style-type: none"> <li>- самостоятельно задавать критерии для анализа рабочей ситуации на основе эталонной ситуации и определять проблему;</li> <li>- планировать текущий контроль своей деятельности в соответствии с заданной технологией деятельности и определенным результатом (целью) или продуктом деятельности;</li> <li>- определять проблему на основе самостоятельно проведенного анализа ситуации;</li> <li>- планировать и оценивать продукт своей деятельности на основе заданных критериев;</li> <li>- определять критерии оценки продукта на основе задачи деятельности;</li> <li>- выбирать способ разрешения проблемы в соответствии с заданными критериями и ставить цель деятельности;</li> <li>- оценивать последствия принятых решений;</li> <li>- анализировать риски</li> </ul>

		<p>(определять степень вероятности и степень влияния на достижение цели) и обосновывать достижимость цели</p> <p><b>знания:</b></p> <ul style="list-style-type: none"> <li>- технологии анализа рабочей ситуации в соответствии с заданными критериями, указывая ее соответствие/несоответствие эталонной ситуации;</li> <li>- принципы осуществления текущего контроля своей деятельности по заданному алгоритму;</li> <li>- способы оценивания продукта своей деятельности по характеристикам</li> </ul>
ОК 4	<p>Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.</p>	<p><b>практический опыт:</b></p> <ul style="list-style-type: none"> <li>- предлагать источник информации определенного типа/конкретный источник для получения недостающей информации и обосновывать свое предложение;</li> <li>- характеризовать произвольно заданный источник информации в соответствии с задачей деятельности;</li> <li>- принимает решение о завершении/продолжении информационного поиска на основе оценки достоверности/непротиворечивости полученной информации;</li> <li>- делать вывод о применимости общей закономерности в конкретных условиях;</li> <li>- делать вывод на основе предоставленных эмпирических или статистических данных</li> </ul> <p><b>умения:</b></p> <ul style="list-style-type: none"> <li>- самостоятельно находить источник информации по заданному вопросу, пользуясь электронным или бумажным каталогом, электронным или бумажным каталогом, справочно-библиографическими пособиями, поисковыми системами Интернета;</li> <li>- указывать недостаток информации, необходимой для решения задачи;</li> <li>- формулировать вопросы, нацеленные на получение недостающей информации;</li> <li>- извлекать информацию по двум и более основаниям из одного или нескольких источников и систематизировать ее в рамках заданной структуры;</li> <li>- делать выводы об объектах, процессах, явлениях на основе сравнительного анализа информации и них по заданным критериям;</li> </ul>

		<ul style="list-style-type: none"> <li>- задавать критерии для сравнительного анализа информации в соответствии с поставленной задачей деятельности</li> </ul> <p><b>знания:</b></p> <ul style="list-style-type: none"> <li>- выделять из содержащего избыточную информацию источника информацию, необходимую для решения задачи;</li> <li>- выделять в источнике информации вывод и/или аргументы</li> </ul>
ОК 5	Использовать информационно-коммуникационные технологии в профессиональной деятельности.	<p><b>практический опыт:</b></p> <ul style="list-style-type: none"> <li>- применять ИКТ при выполнении профессиональных задач</li> </ul> <p><b>умения:</b></p> <ul style="list-style-type: none"> <li>- применять ИКТ при выполнении заданий</li> </ul> <p><b>знания:</b></p> <ul style="list-style-type: none"> <li>- перечислять ИКТ, применяемые в профессиональной деятельности;</li> <li>- ориентироваться в информационно-коммуникационных технологиях, применяемых в профессиональной деятельности</li> </ul>
ОК 6	Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.	<p><b>практический опыт:</b></p> <ul style="list-style-type: none"> <li>- принимать и фиксировать решение по вопросам для группового обсуждения;</li> <li>- фиксировать особые мнения;</li> <li>- использовать приемы выхода из ситуации, когда дискуссия зашла в тупик;</li> <li>- давать сравнительную оценку идей, высказанных участниками группы, относительно цели групповой работы;</li> <li>- самостоятельно готовить средства наглядности;</li> <li>- самостоятельно выбирать жанр монологического высказывания в зависимости от его цели и целевой аудитории;</li> <li>- запрашивать мнение партнера по диалогу;</li> <li>- извлекать из устной речи (монолог, диалог, дискуссия) фактическую и оценочную информацию, определять основную тему, предложения, аргументы, доказательства, выводы, оценки;</li> <li>- самостоятельно определять жанр письменной коммуникации в зависимости от цели;</li> <li>- создавать продукт письменной коммуникации сложной конструкции</li> </ul> <p><b>умения:</b></p> <ul style="list-style-type: none"> <li>- договариваться о процедуре и вопросах для обсуждения в группе в соответствии с</li> </ul>

		<p>поставленной целью деятельности команды (группы);</p> <ul style="list-style-type: none"> <li>- при групповом обсуждении задавать вопросы, проверять адекватность понимания идей других;</li> <li>- соблюдать заданный жанр высказывания (служебный доклад, выступление на совещании, собрании, презентации товара (услуги));</li> <li>- использовать средства наглядности или невербальные средства, направленные на выяснение мнения (позиции);</li> <li>- задавать вопросы, направленные на выяснение фактической информации;</li> <li>- создавать стандартный продукт письменной коммуникации</li> </ul> <p><b>знания:</b></p> <ul style="list-style-type: none"> <li>- правила участия в групповом обсуждении, высказываясь в соответствии с заданной процедурой и по заданному вопросу;</li> <li>- соблюдать нормы публичной речи и регламент, используя паузы для выделения смысловых блоков своей речи;</li> <li>- начинать и заканчивать служебный разговор в соответствии с нормами;</li> <li>- отвечать на вопросы, направленные на выяснение фактической информации;</li> <li>- извлекать из устной речи (монолог, диалог, дискуссия) основное содержание фактической информации</li> </ul>
ОК 7	<p>Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.</p>	<p><b>практический опыт:</b></p> <ul style="list-style-type: none"> <li>- контролировать и отвечать за работу занимающихся</li> </ul> <p><b>умения:</b></p> <ul style="list-style-type: none"> <li>- анализировать работу занимающихся и результат выполненного задания;</li> <li>- оценивать работу и контролировать работу занимающихся</li> </ul> <p><b>знания:</b></p> <ul style="list-style-type: none"> <li>- выполнять поставленные задания, являясь членом группы</li> </ul>
ОК 8	<p>Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.</p>	<p><b>практический опыт:</b></p> <ul style="list-style-type: none"> <li>- анализировать собственные мотивы и внешнюю ситуацию при принятии решений, касающихся своего продвижения</li> </ul> <p><b>умения:</b></p> <ul style="list-style-type: none"> <li>- указывает «точки успеха» и «точки роста»;</li> <li>- указывает причины успехов и неудач в деятельности;</li> </ul>



		<p>- анализировать/формулировать запрос на внутренние ресурсы (знания, умения, навыки, способы деятельности, ценности, установки) для решения профессиональной задачи</p> <p><b>знания:</b></p> <p>- называть трудности, с которыми столкнулся при решении задачи и знать пути их преодоления/избегания в дальнейшей деятельности</p>
ОК 9	Ориентироваться в условиях частой смены технологий в профессиональной деятельности.	<p><b>практический опыт:</b></p> <p>- применять современные технологии в профессиональной деятельности</p> <p><b>умения:</b></p> <p>- сравнивать технологии, применяемые в профессиональной деятельности;</p> <p>- выбирать технологии для своей профессиональной деятельности</p> <p><b>знания:</b></p> <p>- информацию о современных технологиях в профессиональной деятельности</p>
ПК 3.1	Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.	<p><b>иметь практический опыт:</b></p> <p>– участия в выработке требований к программному обеспечению;</p> <p>– участия в проектировании программного обеспечения с использованием специализированных программных пакетов;</p> <p><b>уметь:</b></p> <p>- владеть основными методологиями процессов разработки программного обеспечения;</p> <p>– принимать участие в формировании требований к ПО, выполнять анализ и спецификацию требований, уточнение требований на работающих прототипах;</p> <p>– выполнять системный анализ и проектирование компонент ПО на основе существующих методологий с использованием автоматизированных программных (CASE) средств;</p> <p>– выполнять интеграцию отдельных компонент ПО в единую программную систему, осуществлять их коммуникацию и взаимодействие друг с другом, а также другими программными средствами; – участвовать в разработке и формализованном описании тестовых сценариев, выполнять тестирование ПО</p>
ПК 3.2	Выполнять интеграцию модулей в программную систему.	
ПК 3.3	Выполнять отладку программного продукта с использованием специализированных программных средств.	
ПК 3.4	Осуществлять разработку тестовых наборов и тестовых сценариев.	
ПК 3.5	Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	
ПК 3.6	Разрабатывать технологическую документацию.	

	<p>на основе разработанных спецификаций;</p> <ul style="list-style-type: none"> <li>– выполнять отладку компонент ПО с использованием специализированных программных средств;</li> <li>– принимать участие в предпродажной подготовке, внедрении и сопровождении ПО;</li> <li>– выполнять формализованное описание компонент ПО, формировать техническую и эксплуатационную документацию на основе принятых стандартов с использованием специализированных программных пакетов</li> <li>– использовать методы для получения кода с заданной функциональностью и степенью качества.;</li> <li>– выполнять кодирование компонент ПО на основе разработанных спецификаций и существующих стандартов с использованием современных инструментальных средств разработки (ИСП);</li> <li>- производить инспектирование качества и эффективности программного кода, степени его соответствия стандартам кодирования, выполнять оптимизацию и ревьюирование программного кода с использованием специализированных программных пакетов.</li> </ul> <p><b>знать:</b></p> <ul style="list-style-type: none"> <li>– модели процесса разработки программного обеспечения;</li> <li>– основные принципы процесса разработки программного обеспечения;</li> <li>– основные подходы к интегрированию программных модулей;</li> <li>– основные методы и средства эффективной разработки;</li> <li>– основы верификации и аттестации программного обеспечения;</li> <li>– концепции и реализации программных процессов;</li> <li>– принципы построения, структуры и приёмы работы с инструментальными средствами, поддерживающими создание программного обеспечения;</li> <li>– основные положения метрологии программных продуктов, принципы</li> </ul>
--	---

		<p>построения, проектирования и использования средств для измерения характеристик и параметров программ, программных систем и комплексов;</p> <p>– стандарты качества программного обеспечения;</p> <p>– методы и средства разработки программной документации.</p>
--	--	---

## 2 СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

### 2.1 Объем профессионального модуля и виды учебной работы

Вид учебной работы	Объем часов
<b>Объем образовательной программы профессионального модуля</b>	319
<b>в том числе:</b>	
– теоретическое обучение	114
– практические занятия <i>(если предусмотрено)</i>	98
– лабораторные занятия <i>(если предусмотрено)</i>	
– курсовая работа (проект) <i>(если предусмотрено)</i>	
– самостоятельная работа	87
– консультации	20
– промежуточная аттестация – <i>(форма промежуточной аттестации)</i>	Экзамен

## 2.2 Тематический план и содержание профессионального модуля

Наименование разделов профессионального модуля (ПМ), междисциплинарных курсов (МДК) и тем	Содержание учебного материала, практические занятия, самостоятельная работа обучающихся	Объем часов	Коды компетенций, формирование которых способствует элемент программы
1	2	3	4
<b>ПМ. 03</b>	<b>Участие в интеграции программных модулей</b>	<b>319</b>	ОК 1
<b>МДК .03.01</b>	<b>Технология разработки программного обеспечения</b>	<b>102</b>	ОК 2
<b>Тема 1</b> Определение технологии конструирования программного обеспечения	Содержание Классический жизненный цикл Макетирование Стратегии конструирования ПО Инкрементная модель Быстрая разработка приложений Спиральная модель Компонентно-ориентированная модель Тяжеловесные и облегченные процессы XP-процесс Модели качества процессов конструирования Практические занятия Модели жизненного цикла программного продукта: RAD - модель, спиральная модель	2	ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
<b>Тема 2.</b> Руководство программным проектом	Содержание Процесс руководства проектом Планирование проектных задач Размерно-ориентированные метрики Функционально-ориентированные метрики Выполнение оценки в ходе руководства проектом Выполнение оценки проекта на основе LOC- и FP-метрик Конструктивная модель стоимости Предварительная оценка программного проекта Анализ чувствительности программного проекта Практические занятия Построение модели информационной системы	2	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
<b>Тема 3.</b> Классические методы анализа	Содержание Структурный анализ Методы анализа, ориентированные на структуры данных Метод анализа Джексона Практические занятия Составление плана графика выполнения программного проекта. (GanttProject, MS Visio, MS Excel,	2	

	MS Project)		
<b>Тема 4.</b> Основы проектирования программных систем	Содержание	2	ОК 1
	Особенности процесса синтеза программных систем Особенности этапа проектирования Структурирование системы Моделирование управления Декомпозиция подсистем на модули Модульность Информационная закрытость Связность модуля Сцепление модулей Сложность программной системы Характеристики иерархической структуры программной системы		ОК 2 ОК 3 ОК 4 ОК 5
	Практические занятия Структурные карты Константайна. Структурные карты Джексона	2	ОК 6 ОК 7
<b>Тема 5.</b> Классические методы проектирования	Содержание	2	ОК 8
	Метод структурного проектирования Метод проектирования Джексона		ОК 9
	Практические занятия Разработка приложений основанных на основных принципах ОО П. Объектно - ориентированное программирование. Динамические и статические структуры данных. Использование технологий OLE, COM.	2	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
<b>Тема 6.</b> Структурное тестирование программного обеспечения	Содержание	2	
	Основные понятия и принципы тестирования ПО Особенности тестирования «белого ящика» Способ тестирования базового пути Способы тестирования условий Способ тестирования потоков данных		
	Тестирование циклов Практические занятия Разработка пользовательских интерфейсах на принципах ООП.	2	
<b>Тема 7.</b> Функциональное тестирование программного обеспечения	Содержание	2	ОК 1
	Особенности тестирования «черного ящика» Способ разбиения по эквивалентности Способ анализа граничных значений Способ диаграмм причин-следствий		ОК 2 ОК 3 ОК 4
	Практические занятия Отладка программы. Методом индукции. Методом дедукции.	2	ОК 5 ОК 6
<b>Тема 8.</b>	Содержание	2	ОК 7

Организация процесса тестирования программного обеспечения	Методика тестирования программных систем Тестирование элементов Тестирование интеграции Тестирование правильности Системное тестирование Искусство отладки	2	ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Практические занятия Отладка программы методом обратного прослеживания. Исключения		
<b>Тема 9.</b> Основы объектно-ориентированного представления программных систем	Содержание Принципы объектно-ориентированного представления программных систем Объекты. Классы	4	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Практические занятия Отладка программы с использованием инструментальных средств отладки программного обеспечения	2	
<b>Тема 10.</b> Базис языка визуального моделирования	Содержание Унифицированный язык моделирования Предметы в UML Отношения в UML Диаграммы в UML Механизмы расширения в UML	2	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Практические занятия Разработка тестов и тестирование программы методом «белого ящика»	2	
<b>Тема 11.</b> Статические модели объектно-ориентированных программных систем	Содержание Вершины в диаграммах классов Отношения в диаграммах классов Деревья наследования Примеры диаграмм классов	2	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6
		2	

	Практические занятия Разработка тестов и тестирование программы методом «черного ящика»		ОК 7 ОК 8 ОК 9
<b>Тема 12.</b> Динамические модели объектно-ориентированных программных систем	Содержание Моделирование поведения программной системы Диаграммы схем состояний Диаграммы деятельности Диаграммы взаимодействия Диаграммы сотрудничества Диаграммы последовательности Диаграммы Use Case Спецификация элементов Use Case Пример диаграммы Use Case Построение модели требований Кооперации и паттерны Бизнес-модели	2	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Практические занятия Разработка объектно-ориентированных тестовых вариантов.	2	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
<b>Тема 13.</b> Модели реализации объектно-ориентированных программных систем	Содержание Компонентные диаграммы Использование компонентных диаграмм Основы компонентной объектной модели Работа с СОМ-объектами Диаграммы размещения	2	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4
	Практические занятия Выбор модели надежности программного продукта, определение качества программного продукта.	2	

<b>Тема 14.</b> Метрики объектно-ориентированных программных систем	Содержание Метрические особенности объектно-ориентированных программных систем Эволюция мер связи для объектно-ориентированных программных систем Набор метрик Чидамбера и Кемерера Использование метрик Чидамбера-Кемерера Метрики Лоренца и Кидда Набор метрик Фернандо Абреу Метрики для объектно-ориентированного тестирования	2	ПК 3.5 ПК 3.6
	Практические занятия Тестирование программного продукта.	2	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
<b>Тема 15.</b> Унифицированный процесс разработки объектно-ориентированных ПС	Содержание Эволюционно-инкрементная организация жизненного цикла разработки Управление риском Этапы унифицированного процесса разработки Пример объектно-ориентированной разработки	2	
	Практические занятия Разработка в стиле экстремального программирования	2	
<b>Тема 16.</b> Объектно-ориентированное тестирование	Содержание Расширение области применения объектно-ориентированного тестирования Изменение методики при объектно-ориентированном	1	



	тестировании Проектирование объектно-ориентированных тестовых вариантов Способы тестирования содержания класса Способы тестирования взаимодействия классов		
	Практические занятия Предваряющее тестирование при экстремальной разработке	2	ОК 1 ОК 2
<b>Тема 17.</b> Автоматизация конструирования визуальной модели программной системы	Содержание	1	ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
	Общая характеристика CASE-системы Rational Rose Создание диаграммы Use Case Создание диаграммы последовательности Создание диаграммы классов Создание компонентной диаграммы		2
	Практические занятия Генерация программного кода		
<b>Самостоятельная работа при изучении раздела ПМ 1</b> Подготовка доклада-объяснения с использованием интерактивной доски по темам: «Технология программирования в компании Microsoft.» «Проведение предпроектных исследований; Построение организационно-функциональной структуры компании», «Методы типового проектирования, оценка эффективности использования типовых решений. Классификация Case - средств. Возможности Borland Developer Studio»; «Технологии и стандарты тестирования программного продукта»; «Современные средства отладки программного обеспечения, подготовка к практическим занятиям.»; «Обеспечение качества и надежности в процессе разработки сложных программных средств»		<b>30</b>	
<b>МДК.03.02</b>	Инструментальные средства разработки программного обеспечения	<b>128</b>	ОК 1
<b>Тема 1</b> ИСПП. Основные	Содержание	4	ОК 2 ОК 3
	Порядок разработки.		ОК 4

понятия	Требования к содержанию и документам. История развития ИСРП		ОК 5 ОК 6
	Практические занятия Контекстной диаграммы IDEF0 DIA	2	ОК 7 ОК 8
<b>Тема 2.</b> Документы международного и государственног о стандарта, определяющие состав разработки. RUP.	Содержание	2	ОК 9
	Документы международного и государственного. стандарта, определяющие состав разработки. RUP. Практические занятия Разработка диаграммы декомпозиции	2	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
<b>Тема 3</b> Выработка требований. ТЗ.	Содержание	4	ОК 1
	Выработка требований. ТЗ. Методы и инструменты логического проектирования. Практические занятия Создание диаграмм декомпозиции	4	ОК 2 ОК 3 ОК 4
<b>Тема 4</b> UML. Методы и инструменты	Содержание	4	ОК 5
	UML. Описание функциональности разработки. Методы и инструменты. Практические занятия Создание FEO - диаграммы	2	ОК 6 ОК 7 ОК 8
<b>Тема 5.</b> Создание модели процессов в BPwin (IDEF0).	Содержание	2	ОК 9
	UML. Методы и инструменты. Создание модели процессов в BPwin (IDEF0). Практические занятия Построение концептуальной схемы	2	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4
<b>Тема 6</b> Построение диаграммы классов	Содержание	2	ПК 3.5
	Построение диаграммы классов. Методы, технологии, инструменты Практические занятия Построение логической схемы.	2	ПК 3.6
<b>Тема 7</b> Языки	Содержание	4	ОК 1
	Определение языка разработки, среды реализации, инструментов разработки. Особенности		ОК 2

программирования и языковые системы.	виртуальной среды реализации программ и их учет в разработке ПП. Операционные оболочки микропроцессора.		ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Практические занятия Построение блок-схем	2	
<b>Тема 8</b> Определение инструментов разработки	Содержание	3	
	Системные макросы и их применение в текстах разработки. Инструментальные средства и технологии Windows. MFC. SDK. Практические занятия Построение EPC - диаграмм	2	
<b>Тема 9</b> Процедура физического проектирования	Содержание	4	
	Процедура физического проектирования – порядок, инструменты, ресурсы, документы		
	Практические занятия Построение BPM N - диаграмм.	2	
<b>Тема 10</b> Средства визуального программирования – MS Visual Studio, Borland Delphi	Содержание	2	
	Средства визуального программирования – MS Visual Studio, Borland Delphi		
	Практические занятия Построение диаграмм вариантов использования	4	
<b>Тема 11</b> Средства визуального программирования	Содержание	4	
	Результаты компиляции. Список опций компилятора и компоновщика. Управление компилятором (C++Builder).		
	Практические занятия Построение диаграмм автоматов	2	
<b>Тема 12</b> Подбор и редактирование компонент, разработка компонент. Open TOOLS API	Содержание	4	
	Подбор и редактирование компонент, разработка компонент. Open TOOLS API		
	Практические занятия Построение диаграмм классов анализа	2	

<b>Тема 13</b> Построение интерфейса программы.	Содержание	4	ПК 3.5 ПК 3.6
	Построение интерфейса программы. Принципы разработки инструментария.		
	Практические занятия Построение диаграмм взаимодействия	2	
<b>Тема 14</b> Инструментальные средства и методы построения интерфейса	Содержание	4	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Инструментальные средства и методы построения интерфейса. Добавление действий.		
	Практические занятия Построение диаграмм последовательности	2	
<b>Тема 15</b> Отладка программ	Содержание	4	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	Отладка программ. Инструменты. Методика отладки. Тестирование. Разработка инвариантов и тестовых примеров		
	Практические занятия Построение диаграмм деятельности	2	
<b>Самостоятельная работа при изучении раздела ПМ 2</b> Подготовка доклада-объяснения с использованием интерактивной доски по темам: «Программные продукты для проектирования БД», «Язык SQL»;		<b>37</b>	
<b>МДК.03.03</b>	<b>Документирование и сертификация</b>	<b>89</b>	
<b>Тема 1.</b> Документирование и сертификация ПО	<b>Содержание</b>	5	ОК 1
	Понятие качества. Предмет и задачи курса.		ОК 2
	Управление качеством как фактор успеха предприятия в конкурентной борьбе.		ОК 3
	Стандартизация в системе управления качеством.		ОК 4
	Механизм управления качеством.	ОК 5	
Стандарты ИСО серии 9000.	ОК 6		
Сертификация продукции, услуг и систем менеджмента качества	ОК 7		
<b>Практические занятия</b>	5	ОК 8	
Качество как экономическая категория и объект управления.		ОК 9 ПК 3.1	

<b>Тема 2.</b> Основные понятия и характеристики качества программных средств	<b>Содержание</b> Основные факторы, определяющие качество программных средств. Стандарты, регламентирующие характеристики качества. Метрики характеристик качества программных средств. Особенности измерения и оценивания характеристик качества. Негативные факторы, влияющие на качество. Ресурсы, ограничивающие достижимые характеристики качества	5	ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	<b>Практические занятия</b> Основные понятия и характеристики качества программных средств.	5	
<b>Тема 3.</b> Выбор мер и шкал характеристик качества программных средств	<b>Содержание</b> Принципы выбора характеристик качества. Выбор свойств и атрибутов качества функциональных возможностей. Выбор количественных атрибутов характеристик качества. Выбор качественных атрибутов характеристик качества. Процессы выбора и установления характеристик и мер качества в проектах программных средств.	5	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6
	<b>Практические занятия</b> Выбор мер и шкал характеристик качества программных средств	5	
<b>Тема 4.</b> Стандартизация оценивания технологических процессов жизненного цикла и характеристик качества	<b>Содержание</b> Оценивание уровня зрелости процессов жизненного цикла и обеспечения качества программных средств. Оценивание жизненного цикла программных средств по стандарту ISO 15504. Оценивание качества готового программного продукта по стандарту ISO 14598. Организация и средства для оценивания качества комплексов программ	5	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5
	<b>Практические занятия</b> Стандартизация оценивания технологических процессов жизненного цикла и характеристик качества.	5	
<b>Тема 5.</b> Единая система программной документации	<b>Содержание</b> Общая характеристика ЕСПД. Структура ЕСПД. ГОСТ 19.101. Виды программ и программных документов. ГОСТ 19.102. Стадии разработки. ГОСТ 19.103.	5	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5

	<p>Обозначение программ и программных документов.  ГОСТ 19.105. Общие требования к программным документам. ГОСТ 19.104.  Основные надписи. ГОСТ 19.106.  Требования к программным документам, выполненным печатным способом.  ГОСТ 19.201. Техническое задание.  Требования к содержанию и оформлению. ГОСТ 19.202.  Спецификация. Требования к содержанию и оформлению.  ГОСТ 19.301. Программа и методика испытаний.  Требования к содержанию, оформлению и контролю качества. ГОСТ 19.401.  Текст программы. Требования к содержанию и оформлению. ГОСТ 19.402.  Описание программы. ГОСТ 19.404.  Пояснительная записка.  Требования к содержанию и оформлению. ГОСТ 19.502.  Описание применения.  Требования к содержанию и оформлению.  ГОСТ 19.503. Руководство системного программиста.  Требования к содержанию и оформлению. ГОСТ 19.504. Руководство программиста.  Требования к содержанию и оформлению. ГОСТ 19.505. Руководство оператора.  Требования к содержанию и оформлению. ГОСТ 19.508. Руководство по техническому обслуживанию. Требования к содержанию Общая характеристика ЕСПД. Структура ЕСПД. ГОСТ 19.101.  Виды программ и программных документов. ГОСТ 19.102.  Стадии разработки. ГОСТ 19.103.  Обозначение программ и программных документов.  ГОСТ 19.105. Общие требования к программным документам. ГОСТ 19.104.  Основные надписи. ГОСТ 19.106.  Требования к программным документам, выполненным печатным способом.  ГОСТ 19.201. Техническое задание.  Требования к содержанию и оформлению. ГОСТ 19.202.  Спецификация. Требования к содержанию и оформлению.  ГОСТ 19.301. Программа и методика испытаний.  Требования к содержанию, оформлению и контролю качества. ГОСТ 19.401.  Текст программы. Требования к содержанию и оформлению. ГОСТ 19.402</p>	5	<p>ОК 6  ОК 7  ОК 8  ОК 9  ПК 3.1  ПК 3.2  ПК 3.3  ПК 3.4  ПК 3.5  ПК 3.6</p>
<b>Практические занятия</b>			

	Разработка технического задания на создание программного средства. Разработка технологической документации на программное средство. Разработка эксплуатационной документации на программное средство		
<b>Тема 6.</b> Оценивание характеристик качества программных средств	<b>Содержание</b>	4	
	Оценивание функциональных возможностей. Оценивание надежности функционирования. Оценивание эффективности использования ресурсов ЭВМ. Оценивание практичности. Оценивание сопровождаемости. Оценивание мобильности. Оценивание качества эксплуатационной и технологической документации. Оценивание рисков в жизненном цикле. <u>Интегральное оценивание характеристик качества</u>		
	<b>Практические занятия</b>	5	
	Оценивание качества программного продукта по стандарту ГОСТ 28195		
<b>Самостоятельная работа при изучении раздела ПМ 3</b>		<b>22</b>	
Подготовка доклада-объяснения с использованием интерактивной доски по темам: «положения стандартов в области программного обеспечения», «Качество программного продукта», «Стандарты СММ/СММ1», «метрики оценки программного продукта», «авторское право»			
<b>Консультации</b>		<b>20</b>	
<b>Промежуточная аттестация (экзамен)</b>			
<b>Всего:</b>		<b>319</b>	

### 3 УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

#### 3.1 Материально-техническое обеспечение

Для реализации программы профессионального модуля предусмотрено наличие следующих специальных помещений:

- **Полигон вычислительной техники:** количество посадочных мест – 66 шт., доска меловая – 1 шт., стол для преподавателя – 1 шт., стол для преподавателя компьютерный – 1 шт., стул для преподавателя – 1 шт., стенд «Информация по охране труда и технике безопасности» - 2 шт., таблица по пожарной безопасности – 20 шт.; количество облачных клиентов -12 (LG 23', Intel Xeon CPU E5-2660 v.3 2.6 Ghz ОЗУ 4 Гб, HDD 50 Gb), количество персональных компьютеров -1 (Монитор Acer 17', системный блок: процессор: Intel(R) Pentium(R) Dual CPU E2180 @ 2.00GHz, HDD), 1 экран Projecta, 1 проектор Sanyo PLC; нормативные документы, учебно-методические пособия, дидактические и презентационные материалы

ПО: Microsoft Windows 7 Professional Russian (Academic Open license 47882164), Microsoft Office 2007 RUS (бессрочно 44216302), Winrar (лицензия RUK-web-1355405), Платформа «1С:Предприятие 8.3» (комплект для обучения в высших и средних учебных заведениях - лицензия ОС-01/1693 бессрочная), Adobe Google Chrome(свободное); Adobe Acrobat Reader (свободное); Adobe Flash Player (свободное), Kaspersky Endpoint Security 10 для Windows (Номер лицензии 17E0200430130957417676), Java(TM) 6 Update 26 (свобное), Программный комплекс ГРАНД-Смета 2019 (версия 6.0.2), система программирования КуМир (свободное), справочно-правовая система КонсультантПлюс (Договор № 2020-А 0130), Pascal ABCNet(свободное), справочно-правовая система КонсультантПлюс (Договор № 2020-А 0130), возможность подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации

- **Лаборатория системного и прикладного программирования:** количество посадочных мест – 30 шт., доска меловая – 1 шт., стол для преподавателя – 1 шт., стул для преподавателя – 1 шт., стол для преподавателя компьютерный – 1 шт.; количество персональных компьютеров с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации - 15 (Монитор 17' Acer, системный блок: процессор Intel(R) Pentium(R) Dual CPU E2180 @ 2.00GHz , ОЗУ 2 Гб, HDD 250 Гб), 1 экран Projecta, 1 проектор Sanyo PLC  
ПО: Microsoft Windows 7 Professional Russian (Academic Open license 47882164), Microsoft Office 2007 RUS (бессрочно 44216302), Winrar (Лицензия RUK-web-1355405), Adobe Google Chrome (свободное); Adobe Acrobat Reader (свободное); Adobe Flash Player (свободное), Kaspersky Endpoint Security 10 для Windows (Номер лицензии 17E0200430130957417676), Java(TM) 6 Update 26 (свободное), справочно-правовая система КонсультантПлюс (Договор № 2020-А 0130)

- **Учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:** количество посадочных мест – 82 шт., стол для преподавателя - 1 шт., стол для преподавателя компьютерный – 1 шт., стул для преподавателя – 1 шт., доска переносная (меловая/маркерная) – 1 шт. , количество персональных компьютеров с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации -1 (монитор 17' Samsung, системный блок: Процессор Intel(R) Core(TM)2 Duo CPU E8400 @ 3.00GHz, ОЗУ 2 Гб, HDD 150 Гб), 1 экран Projecta, 1 проектор Sanyo PLC-XU, учебно-методические пособия, дидактические и презентационные материалы, переносные плакаты;  
ПО: Microsoft Windows XP Professional Russian (Academic license - бессрочно), Microsoft Office 2007 RUS (бессрочно 44216302), Winrar (Лицензия RUK-web-1355405), Adobe Google



Chrome (свободное); Adobe Acrobat Reader (свободное); Adobe Flash Player (свободное), Kaspersky Endpoint Security 10 для Windows (Номер лицензии 17E0200430130957417676), Java(TM) 6 Update 26 (свободное), справочно-правовая система КонсультантПлюс (Договор № 2020-А 0130)

### **3.2 Информационное обеспечение реализации программы**

Для реализации программы профессионального модуля библиотечный фонд ВГУЭС укомплектован печатными и электронными изданиями.

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

#### **Основная литература**

1. Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2020. — 235 с. <https://urait.ru/bcode/453640>

2. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для среднего профессионального образования / Е. А. Черткова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 147 с. :<https://urait.ru/bcode/454414>

3. Гниденко, И. Г. Технология разработки программного обеспечения : учебное пособие для среднего профессионального образования / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва : Издательство Юрайт, 2020. — 235 с. <https://urait.ru/bcode/453640>

4. Черткова, Е. А. Программная инженерия. Визуальное моделирование программных систем : учебник для среднего профессионального образования / Е. А. Черткова. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 147 с. :<https://urait.ru/bcode/454414>

5. Кузнецов, И. Н. Документационное обеспечение управления. Документооборот и делопроизводство : учебник и практикум для среднего профессионального образования / И. Н. Кузнецов. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. -462 с. <https://urait.ru/bcode/451242>

6. Сергеев, А. Г. Стандартизация и сертификация : учебник и практикум для среднего профессионального образования / А. Г. Сергеев, В. В. Терегеря. — Москва : Издательство Юрайт, 2020. — 323 с. <https://urait.ru/bcode/451055>

#### **Дополнительная литература**

1. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2020. — 218 с. <https://urait.ru/bcode/456394>

2. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 432 с. <https://urait.ru/bcode/452137>

3. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебное пособие для среднего профессионального образования / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2020. — 218 с. <https://urait.ru/bcode/456394>

4. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 432 с. <https://urait.ru/bcode/452137>

#### **Электронные ресурсы**

1. Научная электронная библиотека eLIBRARY.RU: <http://www.eLIBRARY.RU>

2. Ресурс Цифровые учебные материалы <http://abc.vvsu.ru/>

3. ЭБС «Руконт»: <http://www.rucont.ru/>

4. ЭБС «Юрайт»: <http://www.biblio-online.ru/>

## 4 КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Результаты (освоенные профессиональные компетенции)	Основные показатели оценки результата	Формы и методы контроля и оценки
ПК 3.1 Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.	<ul style="list-style-type: none"> <li>- анализ требований к программному обеспечению;</li> <li>- определение характера взаимодействия компонентов программного обеспечения;</li> <li>- анализ проектной и технической документации на уровне взаимодействия компонент программного обеспечения;</li> <li>- точность и грамотность оформления технологической документации;</li> </ul>	Текущий контроль в форме: - устный опрос; - домашние работы; экспертная оценка защиты лабораторных работ; экспертная оценка результатов тестирования; - экспертная оценка выполнения контрольных работ по темам МДК; -
ПК 3.2 Выполнять интеграцию модулей в программную систему.	<ul style="list-style-type: none"> <li>- определение этапов разработки программного обеспечения;</li> <li>- демонстрация построения концептуальной, логической и физической моделей программного обеспечения и отдельных модулей;</li> <li>- выбор технологии разработки исходного модуля исходя из его назначения;</li> <li>- выбор методов разработки программных модулей;</li> <li>- выбор средств разработки программных модулей; демонстрация навыков модификации программных модулей;</li> </ul>	оценка выполнения самостоятельной работы студентами; Экспертная оценка выполнения практического задания по производственной практике. Защита курсового проекта. Комплексный экзамен по модулю.
ПК 3.3 Выполнять отладку программного продукта с использованием специализированных программных средств.	<ul style="list-style-type: none"> <li>- выявление ошибок в программных модулях;</li> <li>- определение возможности увеличения быстродействия программного продукта;</li> <li>- определение способов и принципов оптимизации;</li> <li>- выбор методов отладки программных модулей и программного продукта;</li> <li>- выбор специализированных средств для отладки программного продукта; демонстрация навыков использования программных средств для отладки программного продукта</li> </ul>	
ПК 3.4 Осуществлять разработку тестовых наборов и тестовых сценариев.	<ul style="list-style-type: none"> <li>- разработка тестовых наборов и тестовых сценариев;</li> <li>- демонстрация устранения ошибок в программных модулях;</li> <li>- демонстрация использования</li> </ul>	

	методов тестирования программного обеспечения; - демонстрация навыков внесения изменения в программные модули для обеспечения качества программного обеспечения; - демонстрация навыков правильного использования инструментальных средств тестирования программных модулей	
ПК 3.5 Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	- выбор методов обеспечения качества и надежности в процессе разработки сложных программных средств. - изложение основных принципов тестирования; - способен производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	
ПК 3.6 Разрабатывать технологическую документацию	- правильность выбора методов средств разработки программной документации; - точность и грамотность оформления технологической документации;	

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

МИНОБРНАУКИ РОССИИ  
ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ЭКОНОМИКИ И СЕРВИСА

КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА  
для проведения текущего контроля и промежуточной аттестации  
по профессиональному модулю

*ПМ.03 Участие в интеграции модулей*

программы подготовки специалистов среднего звена  
*09.02.03 Программирование в компьютерных системах*

Форма обучения: очная

Находка 2020

Контрольно-оценочные средства для проведения текущего контроля и промежуточной аттестации по профессиональному модулю *ПМ.03 «Участие в интеграции модулей»* разработаны в соответствии с требованиями ФГОС СПО по специальности 09.02.03 Программирование в компьютерных системах, утвержденного приказом Минобрнауки России от «28» июля 2014 г. № 804, примерной образовательной программой.

Разработчик(и): Коломийцев А.К., преподаватель ОСПО филиала ФГБОУ ВО «ВГУЭС» в г. Находке.

Рассмотрена на заседании МПЦК от 30 марта 2020 г., протокол № 9

Председатель МПЦК  Фадеева Н.П.

## 1 Общие положения

Результатом освоения профессионального модуля является готовность обучающегося к участию в интеграции программных модулей.

Формой итоговой аттестации по профессиональному модулю является **экзамен (квалификационный)**. Итогом этого экзамена является однозначное решение: «**Вид профессиональной деятельности освоен/не освоен**».

### 1.1 Формы контроля и оценивания элементов профессионального модуля

Таблица 1

Элемент модуля	Форма контроля и оценивания	
	Текущий контроль	Промежуточная аттестация
МДК.03.01 Технология разработки программного обеспечения	- защита лабораторных и практических работ, внеаудиторных работ; - контрольные работы по темам МДК	7 семестр - другая форма контроля 8 семестр - экзамен
МДК.03.02 Инструментальные средства разработки программного обеспечения	- защита лабораторных и практических работ, внеаудиторных работ ; - контрольные работы по темам МДК	7 семестр - другая форма контроля 8 семестр - экзамен
МДК.03.03 Документирование и сертификация	- защита лабораторных и практических работ, внеаудиторных работ; - контрольные работы по темам МДК	8 семестр - дифференцированный зачет
МДК.03.04 1С: Бухгалтерия	- защита лабораторных и практических работ, внеаудиторных работ ; - контрольные работы по темам МДК	8 семестр - дифференцированный зачет
ИН.03.01 Производственная практика по профилю специальности	Защита по каждому из разделов МДК.	8 семестр - дифференцированный зачет
ПМ.01 Участие в интеграции программных модулей		8 семестр - Экзамен (квалификационный)

## 2 Результаты освоения модуля, подлежащие проверке

### 2.1 Профессиональные и общие компетенции

Таблица 2

Профессиональные и общие компетенции	Показатели оценки результата
ПК 3.1 - Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения	Определение характера взаимодействия компонентов программного обеспечения
	Участие в проектировании программного обеспечения с использованием языка объектно-ориентированного моделирования UML
	Точность и грамотность оформления технологической документации
ПК 3.2 - Выполнять интеграцию модулей в программную систему	Владение основными подходами интеграции программных модулей
	Определение этапов разработки программного обеспечения
	Демонстрация построения концептуальной, логической и физической моделей баз данных
	Самостоятельный обоснованный выбор технологии разработки исходного модуля исходя из его назначения
	Самостоятельный обоснованный выбор методов разработки программных модулей
	Демонстрация навыков модификации программных модулей
ПК 3.3 - Выполнять отладку программного продукта с использованием специализированных программных средств	Овладение принципами построения, структурами и приемами работы с инструментальными средствами, поддерживающими отладку программного обеспечения
	Определение возможности увеличения быстродействия программного продукта
	Определение способов и принципов оптимизации
	Выбор методов отладки программных модулей и программного обеспечения
	Выбор специализированных средств для отладки программного обеспечения
	Демонстрация устранения ошибок в программных модулях
ПК 3.4 - Осуществлять разработку тестовых наборов и тестовых сценариев	Тестовый набор для тестирования программного обеспечения на уровне пакета, модуля, класса, компонента
	Тестовый сценарий для тестирования графического интерфейса
	Результаты нагрузочного тестирования программного обеспечения
	Демонстрация навыков правильного

	использования инструментальных средств тестирования программных модулей
ПК 3.5 - Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	Оформление программного кода программного модуля в соответствии с требованиями стилевого оформления
	Оформление программного кода программного обеспечения в соответствии с требованиями стандартов кодирования
	Заключение о соответствии программного модуля стандартам качества программного обеспечения
	Способен производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования
ПК 3.6 - Разрабатывать технологическую документацию	Выполненная документация для пользователя программного обеспечения
	Документация к программному коду
	Контрольные варианты использования программного обеспечения
	Правильность выбора методов средств разработки программной документации
Точность и грамотность оформления технологической документации	
ОК 1 - Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес	Динамика успеваемости по МДК, положительный отзыв руководителя практики.
	Активное посещение учебных занятий и практики, консультаций.
ОК 2 - Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество	Мотивированное обоснование выбора и применения методов и способов решения профессиональных задач.
	Точность, правильность и полнота выполнения профессиональных задач.
	Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями
	Разработка пользовательского интерфейса
ОК 3 - Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность	Демонстрация способности принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.
	Использование специализированных программных средств отладки программных модулей.
	Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.
ОК 4 - Осуществлять поиск и использование информации, необходимой для эффективного выполнения	Обоснованность выбора информационных источников для решения профессиональных задач.



<p>профессиональных задач, профессионального и личностного развития</p>	<p>Оперативность поиска и использования необходимой информации для качественного выполнения профессиональных задач, профессионального и личностного развития.</p> <p>Широта использования различных источников информации, включая электронные.</p> <p>Использование инструментальных средств и графических языков спецификаций для создания компонент проектной и технической документации.</p> <p>Оформление проектной и технической документации в соответствии со стандартами.</p>
<p>ОК 5 - Использовать информационно-коммуникационные технологии в профессиональной деятельности</p>	<p>Оперативность и широта осуществления операций с использованием общего и специализированного программного обеспечения.</p> <p>Создание отдельных компонент.</p> <p>Выполнение спецификаций компонент.</p>
<p>ОК 6 - Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями</p>	<p>Результаты выполнения задания на учебной практике.</p> <p>Положительный отзыв руководителя практики.</p> <p>Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями.</p> <p>Разработка пользовательского Интерфейса.</p> <p>Использование специализированных программных средств отладки программных модулей.</p> <p>Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.</p> <p>Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев.</p> <p>Определение ошибок в программном коде с использованием тестовых наборов.</p>
<p>ОК 7 - Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий</p>	<p>Ответственность за результат выполнения заданий на практике.</p> <p>Способность к самоанализу и коррекции результатов собственной работы.</p> <p>Использование специализированных программных средств отладки программных модулей.</p> <p>Разработка компонент программных модулей с использованием современных</p>

	<p>инструментальных средств и технологий.</p> <p>Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев.</p> <p>Определение ошибок в программном коде с использованием тестовых наборов.</p> <p>Выявление избыточности кода программного продукта и его оптимизация.</p> <p>Анализ оптимизации программного кода модуля.</p>
<p>ОК 8 - Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации</p>	<p>Качество, своевременность и полнота выполнения заданий внеаудиторной самостоятельной работы.</p> <p>Обоснованность постановки целей и задач самообразования.</p> <p>Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями.</p> <p>Разработка пользовательского Интерфейса.</p> <p>Использование специализированных программных средств отладки программных модулей.</p> <p>Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.</p> <p>Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев.</p> <p>Определение ошибок в программном коде с использованием тестовых наборов.</p> <p>Выявление избыточности кода программного продукта и его оптимизация, анализ оптимизации программного кода модуля</p>
<p>ОК 9 - Ориентироваться в условиях частой смены технологий в профессиональной деятельности</p>	<p>Проявление интереса к инновациям в области профессиональной деятельности</p> <p>Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями</p> <p>Разработка пользовательского интерфейса</p> <p>Использование специализированных программных средств отладки программных модулей.</p> <p>Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.</p>

Выполнение тестирования качества  
разработки программных модулей с помощью  
разработанных тестовых наборов и сценариев

	Определение ошибок в программном коде с использованием тестовых наборов.
	Выявление избыточности кода программного продукта и его оптимизация
	Анализ оптимизации программного кода модуля.
ОК 10 - Исполнять воинскую обязанность, в том числе с применением полученных профессиональных знаний (для юношей)	Участие в спортивных мероприятиях колледжа и мероприятиях.
	Соблюдение правил внутреннего распорядка колледжа.

## 2.2 Формируемые знания, умения, практический опыт

В результате изучения профессионального модуля обучающийся должен:

### иметь практический опыт:

ПО1 - участия в выработке требований к программному обеспечению;

ПО2 - участия в проектировании программного обеспечения с использованием специализированных программных пакетов;

### уметь:

У1 - владеть основными методологиями процессов разработки программного обеспечения;

У2 - использовать методы для получения кода с заданной функциональностью и степенью качества;

### знать:

31 - модели процесса разработки программного обеспечения;

32 - основные принципы процесса разработки программного обеспечения;

33 - основные подходы к интегрированию программных модулей;

34 - основные методы и средства эффективной разработки;

35 - основы верификации и аттестации программного обеспечения;

36 - концепции и реализации программных процессов;

37 - принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;

38 - методы организации работы в коллективах разработчиков программного обеспечения;

39 - основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерений характеристик и параметров программ, программных систем и комплексов;

310 - стандарты качества программного обеспечения;

311 - методы и средства разработки программной документации.

## 2.3 Требования к курсовому проекту

Курсовой проект представляет собой комплексную учебно-исследовательскую работу, направленную на решение конкретной управленческой проблемы по итогам изучения учебного курса (учебной дисциплины), нескольких учебных курсов или учебного модуля.

Результатом курсового проектирования является проект, выполненный преимущественно по правилам проектного менеджмента, возможно частично или полностью реализованный. Обоснование методов решения управленческой проблемы осуществляется на основе обзора теории и практики по проблеме и ее анализа с помощью соответствующих инструментов. Выполняется самостоятельно, содержит элементы новизны и творчества.

В рамках модуля ПМ.03 Участие в интеграции программных модулей курсовое проектирование проводится в 8 семестре по дисциплине МДК.03.01 Технология разработки программного обеспечения.

### 2.3.1 Общие требования

Работа студента над курсовым проектом регламентируется рабочим учебным планом, расписанием занятий и документами, указанными в п. 8.

Выбор темы курсового проекта (курсовой работы) студент осуществляет из предложенных преподавателем примерных тем, утвержденных на заседании кафедры в рамках учебной программы дисциплины или отдельно.

Примерная тема уточняется с учетом специфики отрасли, предприятия или методов исследования, согласовывается с преподавателем и вносится в техническое задание.

Техническое задание на курсовое проектирование заполняется студентом и подписывается преподавателем не позднее 1 недели до срока защиты курсового проекта.

Курсовой проект должен быть разработан студентом самостоятельно на основании проведенных исследований и проектирования.

Курсовой проект по дисциплине МДК.03.01 Технология разработки программного обеспечения имеет следующую структуру:

- 1 Титульный лист
- 2 Техническое задание на курсовое проектирование
- 3 Введение
- 4 Анализ требований
- 5 Проектирование
- 6 Реализация
- 7 Тестирование
- 8 Документирование
- 9 Заключение
- 10 Приложения

### 2.3.2 Последовательность работы над курсовым проектом

В работе студента над курсовым проектом можно условно выделить три этапа:

**Первый этап** - подготовительно-организационный, в течение которого уясняется задание, определяются возможные варианты решения поставленной задачи, подбирается необходимая литература, составляется календарный план выполнения курсового проекта.

**Второй этап** - собственно работа над курсовым проектом, т.е. разработка рабочей версии программного приложения (программы, программного комплекса). В результате этого этапа должны быть выполнена структуризация задачи, разработан алгоритм решения задачи, спроектирован интерфейс, выполнена программная реализация алгоритма, проведены тестирование и отладка. Курсовой проект должен быть выполнен в электронном виде и проверен руководителем.

**Третий этап** - оформление пояснительной записки к курсовому проекту.

Руководитель курсового проекта рекомендует студенту необходимую основную литературу, справочные материалы и другие источники по теме курсового проекта, проводит систематические, предусмотренные по расписанию, консультации, контролирует и проверяет работу студента над курсовым проектом, следит за соответствием всех разделов проекта требованиям.

### 2.3.3 Тематика курсового проектирования

31. Разработка автоматизированной системы «Абитуриент».
32. Разработка автоматизированной системы «Библиотека».
33. Разработка автоматизированной системы «Компьютерное тестирование».
34. Разработка автоматизированной системы «Коммунальные платежи».
35. Разработка автоматизированной системы «Расчет заработной платы».
36. Разработка автоматизированной системы «Поликлиника».
37. Разработка автоматизированной системы «Складской учет».
38. Разработка автоматизированной системы «Гостиница».
39. Разработка автоматизированной системы «Прокат дисков».
40. Разработка автоматизированной системы «Автомагазин».
41. Разработка автоматизированной системы «Прокат автомобилей».
42. Разработка информационно-справочной системы по развлекательным предприятиям города.
43. Разработка автоматизированной системы «Бронирование авиабилетов».
44. Разработка информационно-справочной системы «Городской телефонный справочник».
45. Разработка автоматизированной системы «Коллежд».
46. Разработка автоматизированной системы «Торговая организация».
47. Разработка автоматизированной системы «Строительная организация».
48. Разработка автоматизированной системы «Спортивные организации города».
49. Разработка автоматизированной системы «Автомобилестроительное предприятие».
50. Разработка автоматизированной системы «Магазин автозапчастей».
51. Разработка информационной системы представительства туристической фирмы в зарубежной стране.
52. Разработка автоматизированной системы «Аптека».
53. Разработка информационной системы туристического клуба.
54. Разработка информационной системы театра.
55. Разработка автоматизированной системы «Аэропорт».
56. Разработка автоматизированной системы «Институт».
57. Разработка информационной системы зоопарка.
58. Разработка информационной системы ГИБДД.
59. Разработка информационной системы фотоцентра.
60. Разработка информационной системы железнодорожной пассажирской станции.

### 2.3.4 Критерии оценки

- **оценка «отлично»** выставляется обучающемуся при соблюдении всех требований (в том числе и нормативных) и сроков курсового проектирования; если представление работы происходило на высоком уровне; презентационный и раздаточный материал не вызывает существенных замечаний членов комиссии.

- **оценка «хорошо»** выставляется, если при наличии выполненной на высоком уровне пояснительной записки есть замечания по соблюдению нормативных требований и сроков выполнения работы;

- **оценка «удовлетворительно»** выставляется при частичном соблюдении требований к курсовому проекту: суть проблемы раскрыта недостаточно тщательно, в неполном объеме выражена одна из структурных частей работы; есть существенные замечания по соблюдению нормативных требований и сроков исполнения работы;

- **оценка «неудовлетворительно»** если не соблюдена существенная часть требований и сроков выполнения; работа не представлена на публичной защите курсовых проектов.

### 3. Оценка освоения теоретического курса профессионального модуля ПМ.03

#### 3.1. Задания для оценки освоения МДК 03.01 Технология разработки программного обеспечения

##### 3.1.1 Варианты контрольных работ

###### Вариант 1

Создать простой текстовый редактор:

1. В качестве основы использовать компонент QPlainTextEdit.
2. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
  - Текст (Вырезать, Копировать, Вставить)
  - Справка (Справка, О программе)
3. Компонент QPlainTextEdit разместить на всей клиентской области формы.
4. Назначить действия (QAction) на каждый пункт главного меню и соответствующие горячие клавиши.
5. В качестве формы «О программе» использовать компонент QDialog.
6. В качестве компонента для открытия и сохранения файла использовать QFileDialog.
7. Добавить на форму строку состояния QStatusBar.

###### Вариант 2

Создать простой редактор таблицы базы данных:

1. В качестве основы для подключения к базе данных использовать компонент QSqlDatabase.
2. Для компонента QSqlDatabase использовать следующие настройки:
  - Драйвер - QODBC.
  - Настройки ODBC:

DRIVER={Microsoft Access Driver (\*.mdb, \*.accdb)};

FIL={MS Access};

DBQ = путь к БД

- Путь до файла - db.mdb в папке с программой.
3. Создать главное меню программы (компонент QMenuBar) следующей структуры:
    - Файл (Выход)
    - Справка (О программе)
  4. На основной форме разместить компонент представления QTableView для доступа к данным и подключить его к модели данных QSqlTableModel.
  5. Создать виджет QDBNavigator для обеспечения навигации по данным.



6. Связать виджет QDBNavigator с моделью QSqlTableModel с помощью QDataWidgetMapper
7. Назначить действия (QAction) на каждый пункт главного меню и соответствующие горячие клавиши.
8. В качестве модальной формы «О программе» использовать компонент QDialog.

### **Вариант 3**

Создать простой текстовый редактор:

1. В качестве основы использовать компонент QTextEdit.
2. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
  - Текст (Вырезать, Копировать, Вставить)
  - Справка (Справка, О программе)
2. Создать контейнер изображений QList<QPixmap> для действий «Вырезать», «Копировать» и «Вставить».
3. Создать панель инструментов для действий «Вырезать», «Копировать» и «Вставить» (Компонент QToolBar).
4. Назначить изображения из компонента QList<QPixmap> на кнопки панели инструментов QToolBar.
5. Компонент QTextEdit разместить на всей клиентской области формы.
6. Назначить действия (QAction) на каждый пункт главного меню и соответствующие горячие клавиши.
7. В качестве компонента для открытия и сохранения файла использовать QFileDialog.



## Вариант 4

Создать простой графический редактор:

1. В качестве основного компонента использовать QPixmap.
2. В качестве объекта отображения использовать QCircle, из заголовочного файла circle.h.
3. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
  - Справка (Справка, О программе)
4. Назначить действия (QAction) на каждый пункт главного меню и соответствующие горячие клавиши.
5. Соединить слот onCircleClicked() виджета QMainWindow с сигналом объекта QCircle clicked().
6. В качестве компонента для открытия и сохранения файла использовать QFileDialog.
7. Для сохранения изображения использовать компонент QFile.

## Вариант 5

Создать простой редактор таблицы базы данных:

1. В качестве основы для подключения к базе данных использовать компонент QSqlDatabase
2. Для компонента QSqlDatabase использовать следующие настройки:
  - Драйвер - QMYSQL
  - Сервер - localhost
  - Порт - 3307
  - Логин - root
  - Пароль - usbw

(Использовать СУБД MySQL из состава портативного веб-сервера)

3. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Выход)
  - Справка (О программе)
4. На основной форме разместить компоненты QTableView.
5. Создать виджет QDBNavigator для обеспечения навигации по данным.



6. Связать виджет QDBNavigator с моделью QSqlTableModel с помощью QDataWidgetMapper
7. Компонент QDBNavigator разместить внизу формы, а QTableView - на всю клиентскую область формы
8. Назначить действия на каждый пункт главного меню и соответствующие горячие клавиши.
9. В качестве модальной формы «О программе» использовать компонент QDialog.

## Вариант 6

Создать простой графический редактор:

1. В качестве основного компонента использовать QPixmap.
2. В качестве объекта отображения использовать QCircle, из заголовочного файла circle.h
3. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
  - Цвет
  - Справка (Справка, О программе)
4. Назначить действия на каждый пункт главного меню и соответствующие горячие клавиши.
5. Создать кнопку, на которую реализовать возможность выбора цвета (компонент QColorDialog).
6. В качестве компонента для открытия и сохранения файла использовать QFileDialog.
7. Для сохранения изображения использовать QByteArray.

### **Вариант 7**

Создать простой текстовый редактор:

1. В качестве основы использовать компонент QPlainTextEdit.
2. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
  - Шрифт
  - Справка (Справка, О программе)
3. Компонент QPlainTextEdit разместить на всей клиентской области формы.
4. Назначить действия на каждый пункт главного меню и соответствующие горячие клавиши.
5. В качестве формы «О программе» использовать компонент QDialog.
6. В качестве диалогов использовать
  - QFileDialog,
  - QFontDialog.
7. Добавить на форму строку состояния QStatusBar.

### **Вариант 8**

Создать простой интернет браузер:

1. В качестве основного компонента использовать QWebView.
2. Добавить контейнер и панель инструментов (QToolBar)
3. На кнопки панели инструментов назначить соответствующие изображения. Для этого использовать контейнер QList<QPixmap>.
4. Создать строку ввода адреса (QLineEdit).
5. Создать возможность перехода по страницам.

6. Компонент QWebView разместить на всей клиентской области формы.
7. Отображать процесс загрузки страницы с помощью компонента QProgressBar
8. Добавить на форму строку состояния QStatusBar.

### Вариант 9

Создать простой редактор таблицы базы данных:

1. В качестве основы для подключения к базе данных использовать компонент QSqlDatabase
2. В качестве настроек для QSqlDatabase использовать:
  - Драйвер - QSQLITE
  - База данных должна быть размещена на одном уровне с исполняемым файлом под именем db.sqlite
3. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Выход)
  - Справка (О программе)

Назначить действия на каждый пункт главного меню и соответствующие горячие клавиши

4. На основной форме разместить компоненты QTableView.
5. Создать виджет QDBNavigator для обеспечения навигации по данным.



6. Связать виджет QDBNavigator с моделью QSqlTableModel с помощью QDataWidgetMapper.
7. Компонент QDBNavigator разместить внизу формы, а QTableView - на всю клиентскую область формы
8. Добавить на форму строку состояния QStatusBar.

### Вариант 10

Создать простой интернет браузер:

1. В качестве основного компонента использовать QWebView.
2. Добавить контейнер и панель инструментов (QToolBar)
3. На кнопки панели инструментов назначить соответствующие изображения. Для этого использовать компонент QList<QPixmap>.
4. Создать возможность перехода по страницам.
5. Создать главное меню программы (компонент QMenuBar) следующей структуры:
  - Файл (Выход)
  - Навигация (Домой, Назад, Вперед)
  - Справка (Справка, О программе)
6. Создать строку ввода адреса (QLineEdit).

7. Компонент QWebView разместить на всей клиентской области формы.
8. Создать окно загрузки программы QSplashScreen.

### 3.1.2 Типовой тест к дифференцированному зачету

#### 1. Программный продукт - это

- 1) программа для удовлетворения нужд разработчиков, предназначенная для продажи
- 2) комплекс взаимосвязанных программ для решения определенной проблемы массового спроса, подготовленный к реализации как любой вид промышленной продукции
- 3) программная реализация решения задачи на компьютере
- 4) результат разработки какого-либо технического задания

#### 2. Отличительной особенностью программных продуктов является

- 1) системность
- 2) простота
- 3) универсальность
- 4) надежность

#### 3. Сопровождение программного продукта - это

- 1) снабжение программного продукта необходимой документацией
- 2) обнаружение и исправление ошибок
- 3) поддержка работоспособности программного продукта, переход на его новые версии, внесение изменений, исправление обнаруженных ошибок и т.д.
- 4) проверка работоспособности каждой разработанной функции, процедуры, модуля

#### 4. Мобильность программных продуктов - это

- 1) независимость от технического комплекса системы обработки данных, операционной среды, сетевой технологии обработки данных, специфики предметной области и т.п.
- 2) точность выполнения предписанных функций обработки
- 3) способность к внесению изменений
- 4) обеспечение дружелюбного интерфейса для работы конечного пользователя, наличие контекстно-зависимой подсказки или обучающей системы в составе программного средства

#### 5. В условиях существования рынка программных продуктов важными его характеристиками являются:

- 1) количество продаж, наличие программ-конкурентов, длительность продаж
- 2) стоимость, количество продаж, время нахождения на рынке, известность фирмы-разработчика и программы
- 3) внешний интерфейс программы, количество продаж, наличие программ-конкурентов
- 4) модифицируемость, надежность, универсальность, известность фирмы - разработчика

#### 6. Утилитарные программы выполняют роль.

- 1) сервиса
  - 2) клиента
  - 3) сервера
  - 4) программного средства разработки приложений
7. Основными показателями качества программных продуктов является:
- 1) алгоритмическая сложность, полнота и системность функций обработки, объем файлов программы
  - 2) стоимость, количество продаж, наличие программных продуктов аналогичного назначения
  - 3) мобильность, надежность, эффективность, модифицируемость, коммуникативность, учет человеческого фактора
  - 4) модифицируемость, надежность, наличие программных продуктов аналогичного назначения
8. Функциональные задачи - это
- 1) задачи, требующие решения при реализации функций управления в рамках информационных систем предметных областей
  - 2) основа для разработки сервисных средств ПО (утилиты, библиотеки)
  - 3) совокупность связанных между собой функций и задач управления, с помощью которых достигается выполнение поставленных целей
  - 4) задачи, которые ставятся и решаются при организации технологического процесса обработки информации на компьютере
9. Алгоритм - это
- 1) комплекс математических вычислений для решения задачи
  - 2) последовательность команд, предназначенных для решения задач
  - 3) программная реализация на компьютере решения определенной задачи
  - 4) результат интеллектуального труда, для которого характерно творчество
10. При индивидуальной разработке фирма-разработчик создает программный продукт для.
- 1) конкретного заказчика
  - 2) массового использования
  - 3) внедрения в специальные организации
  - 4) для удовлетворения собственных нужд
11. Модифицируемость программных продуктов означает.
- 1) независимость от технического комплекса системы обработки данных, операционной среды, сетевой технологии обработки данных, специфики предметной области и т.п.
  - 2) точность выполнения предписанных функций обработки
  - 3) способность к внесению изменений, например расширение функций обработки, переход на другую техническую базу обработки и т.п.
  - 4) обеспечение дружественного интерфейса для работы конечного пользователя, наличие контекстно-

зависимой подсказки или обучающей системы в составе программного средства

12. Жизненный цикл программы - это

- 1) временной интервал, начиная с момента замысла программы и кончая прекращением всех видов его пользований
- 2) временной интервал, начиная с момента введения программы в эксплуатацию
- 3) промежуток времени, который определяет наиболее эффективное использование создаваемой программы
- 4) временная характеристика разработки программного продукта

13. Программы малого Жизненного Цикла - это программы

1. когда время разработки программы значительно меньше времени эксплуатации программы
2. когда время разработки программы значительно больше времени использования программы
3. когда время разработки программы равно времени эксплуатации программы
4. нет правильного ответа

14. Выбрать правильный ответ

- a. На этапе сбора и анализа требований заказчик должен выяснить, прежде всего, необходимость обеспечения безопасности системы и данных
- b. На этапе сбора и анализа требований заказчик должен выяснить, прежде всего, функции, которые должен выполнять программный продукт
- c. На этапе сбора и анализа требований заказчик должен выяснить, прежде всего, сроки написания программы
- d. На этапе сбора и анализа требований заказчик должен собрать литературу по разрабатываемому программному продукту

15. Самая распространенная модель Жизненного цикла программного продукта это

- 1) итерационная
- 2) V - образная
- 2) спиральная
- 3) каскадная

16. Классическая модель ЖЦПО характеризуется следующими основными особенностями

1. последовательным выполнением входящих в ее состав этапов
2. наличием обратных связей между этапами
3. отсутствием временного перекрытия этапов
4. отсутствием (или определенным ограничением) возврата к предыдущим этапам
5. наличием результата после каждого этапа разработки

17. Выберите правильную последовательность этапов спиральной модели жизненного цикла программного продукта:

- 1) техническое проектирование, сопровождение ПП, сбор и анализ требований заказчика, кодирование, уточнение функциональных характеристик, тестирование и отладка
- 2) кодирование, техническое проектирование, уточнение функциональных характеристик, сопровождение 11, тестирование и отладка
- 3) кодирование, техническое проектирование, уточнение функциональных характеристик, тестирование и отладка
- 4) определение требований, анализ, реализация и тестирование, внедрение

18. V - образная модель ЖЦ разработки ПО предполагает:

1. отсутствие временного перекрытия этапов
2. наличие обратной связи
3. возможность сокращения времени разработки ПО
4. возможность увеличения жизненного цикла программного продукта

19. На втором этапе каскадной модели ЖЦ разработки ПО (Требования ПО) осуществляется...

1. составление концептуальной структуры системы
2. определение функциональности программного компонента
3. составление детальной спецификации архитектуры системы
4. составление набора тестовых данных

20. Происходит ли интеграция отдельных компонент системы при разработке ПП по каскадной модели?

1. да
2. нет

21. Проверка корректности требований при использовании V - образной модели ЖЦ разработки ПО осуществляется.

1. после каждого этапа разработки
2. после разработки всей системы
3. после разработки черновой версии системы
4. после разработки набора тестовых данных

22. Выберите правильную последовательность этапов жизненного цикла программного продукта:

- 1) техническое проектирование, сопровождение ИИ, сбор и анализ требований заказчика, кодирование, уточнение функциональных характеристик, тестирование и отладка
- 2) сбор и анализ требований, проектирование системы, кодирование, создание программной документации, сопровождение
- 3) кодирование, сбор и анализ требований заказчика, техническое проектирование, уточнение функциональных характеристик, сопровождение 11, тестирование и отладка
- 4) сбор и анализ требований заказчика, уточнение функциональных характеристик, техническое проектирование, кодирование, тестирование и отладка, сопровождение 11

23. Во вспомогательные процессы ЖЦ программного продукта входит:

- 1) документирование, верификация, аттестация, обеспечение качества, совместная оценка, разрешение проблем, аудит
- 2) управление, создание инфраструктуры, усовершенствование, обучение
- 3) разработка, приобретение, поставка, эксплуатация, сопровождение
- 4) кодирование, тестирование, сопровождение

24. Метод получения информации при проектировании программного продукта путем анализа материала подразумевает:

- 1) изучение материала, с которым будет осуществляться работа с использованием данного ШИ
- 2) изучение работы одного из исполнителей с учетом того, что другие исполнители будут выполнять те же действия и операции
- 3) накопление опыта разработки программного продукта
- 4) накопление информации в том случае, если были получены противоречивые сведения

25. Одним из достоинств классического жизненного цикла программного продукта является

- 1) дает план и временной график по всем этапам проекта
- 2) в конце всей работы заказчику будут доступны результаты проекта
- 3) системный анализ каждого элемента программы
- 4) отсутствие временного перекрытия этапов разработки программного продукта

26. Итерационная модель ЖЦПО характеризуется следующими основными особенностями:

1. последовательным выполнением входящих в ее состав этапов
2. наличием обратных связей между этапами
3. отсутствием временного перекрытия этапов
4. отсутствием (или определенным ограничением) возврата к предыдущим этапам
5. возможность проведение корректировки после каждого этапа

27. В конце каждого витка спирали спиральной модели ЖЦ разработки ПО получаем...

1. готовый программный продукт
2. одну версию программного продукта
3. версию программного продукта с набором тестовых данных
4. черновую модель программного продукта

28. Спиральная модель ЖЦ разработки ПО предполагает:

1. отсутствие временного перекрытия этапов
2. наличие обратной связи
3. возможность сокращения времени разработки ПО



29. На втором этапе каскадной модели ЖЦ разработки ПО (Требования к ПО) осуществляется.
1. определение функциональности программного компонента
  2. составление детальной спецификации архитектуры системы
  3. составление концептуальной структуры системы
  4. написание программного кода
30. Происходит ли интеграция отдельных компонент системы при разработке ПП по экстремальной модели ЖЦ?
1. да
  2. нет
31. Какую модель жизненного цикла разработки ПО целесообразнее использовать, если нет четко определенных требований к будущей системе?
- 1) каскадную
  - 2) спиральную
  - 3) V - образную
  - 4) итерационную
32. В каких годах разработана основная часть документов единой системы программной документации?
- 1) 60-70-х гг
  - 2) 70-80-х гг
  - 3) 80-90-х гг
  - 4) 60-90-х гг
33. На сколько частей можно разделить ЕСПД?
- 1) 6
  - 2) 8
  - 3) 5
  - 4) 10
34. Группа «0» ЕСПД указывает на ...
- 1) резервные группы
  - 2) общее положение
  - 3) основополагающие стандарты
  - 4) прочие стандарты
35. Обозначение стандарта ЕСПД должно состоять из трех частей, где первое число обозначает.
- 1) присвоение к классу стандартов ЕСПД
  - 2) код классификационной группы стандарта

3) год регистрации стандарта

4) номер документа в реестре

36. Техническое задание - это

1) набор правил, по которым строится НИ

2) задание, которое необходимо выполнить на ПК по техническим характеристикам

3) набор правил эксплуатации программного продукта

4) совокупность требований к программным средствам, которые могут использоваться как критерий проверки и приемки разработанного ПП

37. ГОСТ 19.102-77 называется:

1) «Правила внесения изменений в программные документы, выполняемые непечатным способом»

2) «Стадии разработки»

3) «Техническое задание. Требования к содержанию и оформлению»

4) «Основные надписи»

38. Техническое задание, как одно из стадий разработки, состоит из трех частей. Каких?

1) научно-исследовательская работа, разработка эскизного проекта, разработка технического проекта

2) разработка программной документации, утверждение эскизного и технического проектов, подготовка и передача программы

3) обоснование необходимой разработки программы, научно-исследовательская работа, разработка и утверждение технического задания

4) разработка программной документации, утверждение выбранных методов разработки, утверждение технического проекта, кодирование

39. Одной из составных частей рабочего проекта является

1) разработка программ и программной документации

2) разработка технического задания;

3) выбор языка программирования

4) разработка технического проекта

40. Системное программное обеспечение - это

1) комплекс взаимосвязанных программ для решения задач определенного класса конкретной предметной области

2) совокупность программ и программных комплексов для обеспечения работы компьютера и сетей ЭВМ

3) совокупность программ и программных комплексов, обеспечивающих технологию разработки, отладки и внедрения создаваемых программных продуктов

41. Предметная (прикладная) область - это

1) проблема, подлежащая решению

- 2) совокупность связанных между собой функций, задач управления, с помощью которых достигается выполнение поставленных целей
- 3) программная реализация решения задачи
- 4) точная формулировка решения задачи на компьютере с описанием входной и выходной информации

42. Задача - это

- 1) проблема, подлежащая решению
- 2) совокупность связанных между собой функций, задач управления, с помощью которых достигается выполнение поставленных целей
- 3) программная реализация решения задачи
- 4) точная формулировка решения задачи на компьютере с описанием входной и выходной информации

43. Программное средство - это

- 1) программа для удовлетворения нужд разработчиков, предназначенная для продажи
- 2) программа, предназначенная для многократного применения на различных объектах и разработанная любым способом
- 3) программная реализация решения задачи на компьютере
- 4) результат разработки какого-либо технического задания

44. Технологические задачи - это

- 1) задачи, требующие решения при реализации функций управления в рамках информационных систем предметных областей
- 2) основа для разработки сервисных средств ПО (утилиты, библиотеки)
- 3) совокупность связанных между собой функций и задач управления, с помощью которых достигается выполнение поставленных целей
- 4) задачи, которые ставятся и решаются при организации технологического процесса обработки информации на компьютере

45. Дайте расшифровку аббревиатуре ЕСПД.

- 1) Единая схема проектирования документов
- 2) Единая система программной документации
- 3) Единая схема программных документов
- 4) Единственная система программной документации

1. На современном этапе выделяют 2 основных подхода к проектированию ПП. Какие?

- 1) структурный и процедурный
- 2) объектно-ориентированный и структурный
- 3) метод проектирования Джексона и объектно-ориентированный
- 4) иерархический и сетевой

2. Методами структурного проектирования являются

1) модульное программирование, нисходящее проектирование, кодирование и тестирование, структурное проектирование;

2) интегрированное и модульное проектирование;

3) функционально - ориентированное и объектно-ориентированное проектирование

4) структурное программирование, модульное проектирование, тестирование и кодирование

3. Что не использует структурный подход проектирования программного продукта?

1) диаграммы декомпозиции

2) интегрированную структуру данных предметной области

3) структурные схемы

4) анализ предметной области

4. Объектно-ориентированный подход проектирования программного продукта основан на:

1) проектировании

2) кодировании и тестировании

3) создании иерархии классов, наследовании свойств объектов и методов их обработки

4) выделении классов объектов

5. Проектирование - это

1) итерационный процесс, при помощи которого требования к программным средствам транслируются в инженерное представление

2) процесс построения модели будущего программного средства

3) инженерное представление программного продукта на каком-либо языке программирования

4) представление программного продукта совокупностью объектов и их свойств

6. Дополните фразу: предварительное проектирование программного продукта формирует...

1) уточнение абстракций и добавляет подробности алгоритмического уровня

2) абстракцию архитектурного уровня

3) идентификацию подсистемы и определение основных принципов управления подсистемами

4) набор тестовых данных

7. Какие модели можно использовать при структурировании системы?

1) модель абстракционной машины, трехуровневую модель, модель хранилища данных, модель клиент-сервер

2) модель событийного управления, модель хранилища данных, модель потока данных, трехуровневую модель

3) модель объекта, модель централизованного управления, модель хранилища данных, модель абстракционной машины

4) модель объекта, модель централизованного управления, модель абстрактной машины

8. Назовите виды моделей управления.

- 1) модель потока данных и модель хранилища данных
- 2) модель клиент-сервер и модель управления прерываниями
- 3) модель централизованного и событийного управления
- 4) модель централизованного и периферийного управления

9. При разбиении программного средства на отдельные модули можно выделить 2 модели:

- 1) модель потока данных и модель событий
- 2) модель потока данных и модель объекта
- 3) модель объекта и модель управления
- 4) модель управления и модель событий

10. В основе модели потока данных лежит -

- 1) сцепление компонентов
- 2) разделение данных
- 3) разбиение по функциям
- 4) выделение отдельных компонентов и их свойств

11. К классическим методам проектирования ПС ориентированных на процедурную реализацию относят:

- 1) структурный метод проектирования и метод проектирования Джексона
- 2) метод проектирования Джексона и объектно-ориентированный метод проектирования
- 3) объектно-ориентированный метод проектирования и структурный метод проектирования
- 4) иерархический и структурный методы

12. Структурный метод проектирования ПП основан на:

- 1) разбиении единой системы на автономные объекты реального мира
- 2) выделении объектов и их методов
- 3) разбиении всей программы на отдельные логические части
- 4) последовательной декомпозиции всей системы на отдельные компоненты

13. Исходными данными для структурного метода проектирования ПП являются

- 1) логические компоненты, составляющие программное средство
- 2) отдельные компоненты модели анализа программных средств
- 3) специальные компоненты, имеющие в основе собственный набор данных
- 4) объекты системы, их свойства и методы

16. Проектирование для потоков типа преобразования состоит из

- 1) 3 уровней

- 2) 5 уровней
- 3) 4 уровней
- 4) 7 уровней

17. Модуль — это...

- 1) самостоятельная часть программы, имеющая определенное назначение и обеспечивающая заданные функции обработки автономно от других программ
- 2) упорядоченный набор команд, обеспечивающий выполнение определенных функций
- 3) алгоритм построения программного продукта

18. Головной модуль -

- 1) обеспечивает вызов других модулей на обработку
- 2) управляет запуском программного продукта
- 3) выполняет функции обработки
- 4) осуществляет обслуживающие функции

19. Синтаксическая спецификация модуля программного продукта позволяет

- 1) построить на используемом языке программирования синтаксически правильное обращение к модулю
- 2) описать семантику функций, выполняемых этим модулем по каждому из его входов
- 3) описать древовидную структуру модуля

20. Метод восходящей разработки программного продукта заключается в

- 1) первоначальном построении модульной структуры в виде дерева затем проектируется каждый модуль в отдельности начиная с нижнего уровня
- 2) первоначальном построении модульной структуры в виде дерева затем проектируется каждый модуль в отдельности начиная с головного уровня
- 3) проектировании всех модулей одновременно и последующая их сборка в единую систему

21. Конструктивный подход в разработке ШП основывается на

- 1) восходящей разработке ШП
- 2) архитектурной разработке ШП
- 3) нисходящей разработке ШП
- 4) является комбинированным способом проектирования

22. Спецификация программного модуля содержит

- 1) общие требования к ШП
- 2) функциональную и синтаксическую спецификацию
- 3) требования к каждому модулю ШП

23. Для контроля структуры программы можно использовать

- 1) смежный контроль
  - 2) тестовый контроль
  - 3) итоговый контроль
  - 4) сквозной контроль
24. В основе структуры программного продукта лежит
- 1) функциональная организация продукта и его функций
  - 2) модульная организация продукта и его функций
  - 3) пошаговое программирование всех функций продукта
25. Рабочий модуль
- 1) обеспечивает вызов других модулей на обработку
  - 2) управляет запуском программного продукта
  - 3) выполняют функции обработки
- а) осуществляют обслуживающие функции
26. Модульную структуру Ш1 можно представить
- 1) в виде сетевой структуры
  - 2) в виде древовидной структуры
  - 3) в реляционной структуры
27. При нисходящем тестировании первым тестируется
- 1) управляющий модуль
  - 2) головной модуль
  - 3) рабочий модуль
  - 4) исполняемый модуль
28. При конструктивном подходе к разработке Ш1 головной модуль программируется исходя из
- 1) спецификаций модуля
  - 2) спецификаций программы в целом
  - 3) общих требований к программному продукту
29. В классическом методе нисходящей разработки программного продукта рекомендуется
- 1) сначала запрограммировать весь программный продукт, а затем начинать нисходящее их тестирование
  - 2) сначала все модули разрабатываемой программы запрограммировать, а затем начинать нисходящее их тестирование
  - 3) сначала запрограммировать весь программный продукт, а затем начинать детально прорабатывать каждый модуль

30. Статический контроль состоит в

- 1) мысленном прокручивании структуры программы при выполнении заранее разработанных тестов
- 2) контроле со стороны разработчиков архитектуры и внешнего описания ПС
- 3) оценке структуры программы с точки зрения хорошо ли программа разбита на модули

31. Цели структуризации программного продукта.

- 1) контролировать трудозатраты и стоимость проектных работ
- 2) разрозненное выполнение отдельных функций программного продукта
- 3) распределить работы по исполнителям, обеспечив приемлемую их загрузку и требуемые сроки разработки программных продуктов

32. Управляющий модуль -

- 1) обеспечивает вызов других модулей на обработку
- 2) управляет запуском программного продукта
- 3) выполняют функции обработки
- 4) осуществляют обслуживающие функции

33. Функциональная спецификация модуля позволяет

- 1) построить на используемом языке программирования синтаксически правильное обращение к модулю
- 2) описать семантику функций, выполняемых этим модулем по каждому из его входов
- 3) описать древовидную структуру модуля

34. Метод нисходящей разработки программного продукта заключается в

- 1) проектировании всех модулей одновременно и последующая их сборка в единую систему
- 2) первоначальном построении модульной структуры в виде дерева затем проектируется каждый модуль в отдельности начиная с нижнего уровня
- 3) первоначальном построении модульной структуры в виде дерева затем проектируется каждый модуль в отдельности начиная с головного уровня

35. Архитектурный подход в разработке ШП основывается на

- 1) восходящей разработке ШП
- 2) нисходящей разработке ШП
- 3) является комбинированным способом проектирования

36. В рамках конструктивного подхода сначала реализуются

- 1) простейшие модули
- 2) более сложные модули
- 3) разрабатывается программа целиком

37. Для контроля структуры программы можно использовать



- 1) смежный контроль
- 2) тестовый контроль
- 3) модульный контроль
- 4) сквозной контроль

38. Модули бывают

- 1) главные, вспомогательные, сервисные, управляющие
- 2) головные, управляющие, рабочие, сервисные
- 3) главные, побочные, рабочие
- 4) первичные, вторичные, локальные, глобальные

39. Сервисный модуль

- 1) обеспечивает вызов других модулей на обработку
- 2) управляет запуском программного продукта
- 3) выполняет функции обработки
- 4) осуществляет обслуживающие функции

40. Назовите методы разработки структуры программного продукта.

- 1) нисходящий, восходящий, конструктивный и архитектурный метод разработки
- 2) разработка сверху вниз и восходящая разработка
- 3) нисходящая, восходящая, детальная разработка
- 4) конструктивна, восходящая, нисходящая разработка

41. При восходящем тестировании программного продукта для каждого модуля необходимо

- 1) создавать отдельный набор тестирующих данных и проверять взаимодействие модуля с остальными модулями и головной программой
- 2) создавать ведущую программу, которая должна подготовить для тестируемого модуля необходимое состояние информационной среды и произвести требуемое обращение к нему
- 3) создавать дополнительный модуль, который тестирует исходный модуль

42. Важным при архитектурном подходе проектирования программного продукта является

- 1) разработка конкретной программы
- 2) разработка конкретной функции программы
- 3) повышение уровня используемого языка программирования
- 4) возможность разработки архитектуры программного продукта после проектирования основных модулей

43. Достоинством целенаправленной конструктивной реализации является то,

- 1) что на ранней стадии создается протестированный программный продукт

- 2) что уже на ранней стадии создается работающий вариант разрабатываемой программы
- 3) что Ш1 создается поэтапно
44. Сквозной контроль - это
  - 1) мысленное прокручивание структуры программы при выполнении заранее разработанных тестов
  - 2) контроль со стороны разработчиков архитектуры и внешнего описания ПС
  - 3) проверка каждого действия разработчика
  - 4) проверка структуры программы, набора разработанных тестов и последующее применение тестов к программе
45. Какую связь устанавливает пользовательский интерфейс?
  - 1) серверную
  - 2) доменную
  - 3) клиент-серверную
  - 4) клиентскую
46. Выберите правильную характеристику системного программиста.
  - 1) системные программисты знают тонкости построения ПС и могут ее модифицировать
  - 2) системные программисты умеют строить алгоритмы и на основе их реализовывать программу
  - 3) системные программисты способны разрабатывать базовые методы и средства оснащения ПО
  - 4) системные программисты не имеют опыт работы с ПК
47. Каким базовым требованиям должен отвечать интерфейс, чтобы он был удобным?
  - 1) понятность, эстетичность, устойчивость к неполадкам
  - 2) простота, надежность, предсказуемость, адаптивность, стандартность
  - 3) легкая восприимчивость пользователями, несхожесть с другими интерфейсами
  - 4) предсказуемость, модифицируемость, адаптивность, эффективность
48. Интерактивный режим работы с программами заключается в том, что.
  - 1) действия пользователей ограничивает задание
  - 2) пользователь вводит команды и получает результат
  - 3) оба варианта правильны
49. Одной из особенностей диалогового режима работы программы является:
  - 1) предназначен для непрограммистов
  - 2) широко применяется в современных ИС
  - 3) программа занимает мало ресурса, в том числе и памяти
  - 4) применяется в MSDOS
50. Пакетный режим работы программ предназначен для.

- 1) работы с самой программой
  - 2) коммерческих целей
  - 3) работы системных программистов и администраторов сети
1. Язык программирования - это.
    - 1) некоторая структура, в которой указаны правила синтаксиса и семантики
    - 2) множество текстов некоторого алфавита, удовлетворяющих правилам синтаксиса и семантики
    - 3) шаблон построения программных продуктов, с заранее указанными правилами синтаксиса и семантики
    - 4) совокупность правил описания алгоритма
  2. Алфавит языка программирования включает в себя
    - 1) буквы и специальные знаки
    - 2) зарезервированные слова
    - 3) оба варианта правильны
  3. Семантика языка программирования - это...
    - 1) правила, определяющие какие операции, и в какой последовательности должна выполнять ЭВМ
    - 2) правила, определяющие какие операции должна выполнять ЭВМ
    - 3) набор правил, которым должна удовлетворять любая программа
    - 4) описание допустимых элементов, используемых языком программирования
  4. Абстракция объектов -
    - 1) представление данных, позволяющее использовать их, не зная подробности их создания и организации
    - 2) происходит в виде процедур, функций, когда фрагмент программы может быть использован другими пользователями, знающими данный фрагмент
    - 3) правильного варианта нет
  5. Одной из сравнительных характеристик языка программирования является уровень языка, который характеризуется
    - 1) разнообразием создаваемых программ
    - 2) сложностью задач с использованием данного языка программирования
    - 3) свойствами совокупности понятий, служащих для описания данного языка программирования
    - 4) сложностью тестирования задач, разработанных на данном языке программирования
  6. Гибкость языка программирования обеспечивает -
    - 1) описание задачи с использованием определенной предметной области
    - 2) легкость выражения в данном языке программирования, необходимое для решения задачи
    - 3) независимость языка от других программных и аппаратных средств

4) возможность тестирования программы на другом языке программирования

7. Полнота языка программирования обеспечивает -

- 1) легкость восприятия пользователем программы
- 2) легкость понимания семантики языков конструкций
- 3) простоту разработки программного продукта
- 4) описание задачи с использованием определенной предметной области

8. Что такое консольное приложение?

- 1) это монитор и клавиатура, рассматриваемые как единое устройство
- 2) устройством вывода
- 3) программа, предназначенная для работы в операционной системе MS-DOS
- 4) обычное приложение Delphi

9. Модульная программа - это программа, в которой...

- 1) модульную часть логической структуры можно изменить, не внося изменений в основную часть программы
- 2) модульную часть программы можно оставить без изменений, а внести изменения в саму программу
- 3) модульная часть функционирует независимо от программы

10. Модуль - это

- 1) отдельная единица, которая представляет собой «шаблон» для построения программных продуктов
- 2) отдельно компилируемая часть программы
- 3) одна из составляющих программного средства, которая функционирует при подключении к программе
- 4) именованная часть программы, которая создается с использованием языка программирования и его настроек

11. Основным свойством модуля является:

- 1) надежность
- 2) независимость
- 3) взаимодействие с программными средствами
- 4) наличие внешних связей

12. Логика модуля означает

- 1) что делает модуль
- 2) как модуль связан с другими модулями программы
- 3) наличие внутренних и внешних потоков информации
- 4) как реализован модуль

13. Внутренняя связь модуля характеризуется.

- 1) понятием прочности модуля
- 2) понятием надежности модуля
- 3) понятием взаимозависимости модуля
- 4) логикой модуля

14. К уровням прочности модуля можно отнести:

- 1) прочность по совпадению, процедурно прочные модули, функционально прочные модули
- 2) прочность по классу, информационные модули
- 3) оба варианта правильны
- 4) прочность по объектам и используемым локальным и глобальным переменным

15. Внешние связи характеризуются степенью...

- 1) независимости модуля по отношению к данным
- 2) зависимости модуля по отношению к данным
- 3) прочностью модуля по отношению к данным

16. Загрузочный модуль - это.

- 1) отдельная программа, независима от других выполняемых программ
- 2) выполняемый \*.exe модуль, фактически отдельная программа
- 3) обычный текстовый файл с нужным расширением
- 4) средство языка программирования служащее для увеличения уровня языка программирования

17. Подпрограмма - это.

- 1) средство языка программирования, служащее для увеличения уровня языка программирования
- 2) набор откомпилированных программ, собранных в специально форматированный файл
- 3) практическая часть текста программы, которая подставляется при каждом вызове, увеличивая ее размер
- 4) обычный текстовый файл с нужным расширением

18. Объектный модуль - это.

- 1) отдельная программа, независима от других выполняемых программ
- 2) модуль специальной структуры, созданный при компиляции программы
- 3) обычный текстовый файл с нужным расширением
- 4) средство языка программирования служащее для увеличения уровня языка программирования

19. Загрузочный модуль - это.

- 1) отдельная программа, независима от других выполняемых программ
- 2) выполняемый \*.exe модуль, фактически отдельная программа

- 3) обычный текстовый файл с нужным расширением
- 4) средство языка программирования служащее для увеличения уровня языка программирования
20. Модуль, выполняющий (реализующий) одну какую-либо определенную функцию называется...
  - 1) информационно прочным
  - 2) объектным
  - 3) независимым
  - 4) функционально прочным
21. Сцепление модуля - это
  - 1) мера его зависимости по данным от других модулей
  - 2) набор откомпилированных программ, собранных в специально форматированный файл
  - 3) практическая часть текста программы, которая подставляется при каждом вызове, увеличивая ее размер
  - 4) степень взаимосвязи локальных переменным модуля и глобальных переменных основной программы
22. Сцепление модулей по общей области означает что.
  - 1) несколько модулей используют одну и ту же область памяти
  - 2) один из них имеет прямые ссылки на состав другого модуля
  - 3) один модуль имеет ссылку на переменные другого модуля
  - 4) несколько модулей имеют ссылку на переменные другого модуля
23. Сцепление модулей по содержимому означает что.
  - 1) несколько модулей используют одну и ту же область памяти
  - 2) несколько модулей имеют ссылку на переменные другого модуля
  - 3) один из них имеет прямые ссылки на состав другого модуля
  - 4) один модуль имеет ссылку на переменные другого модуля
24. Объектно-ориентированная декомпозиция обеспечивает.
  - 1) разбиение единой системы на автономные объекты
  - 2) разбиение единой системы на объекты реального мира
  - 3) оба варианта правильны
25. Объектно-ориентированное программирование строится на 3-х принципах:
  - 1) на абстрагировании, инкапсуляции, модульности
  - 2) на структурированности, абстрагировании, кодировании
  - 3) на инкапсуляции, логике, модульности
26. Абстракция концентрирует внимание ...
  - 1) на внутренние представления объекта

2) на внешние представления объекта

3) на определении способности подвергать декомпозиции

27. Инструментами иерархической структуры ООП программы являются:

1) структуры из классов и объектов

2) структуры из подпрограмм и классов

3) структуры из наследования классов

28. Выбрать правильный ответ.

1) объект - структурная единица, составляющая в целом программу

2) объект - это именованная часть любой программы

3) объект - это конкретное представление отдельной абстракции

29. Состояние объекта -

1) это свойство, отличающее один объект от других

2) характеризуется перечнем функций и свойств данного объекта

3) характеризует то, как данный объект «общается» с другими объектами

4) описывает связь одного объекта программы с другими объектами

30. Оптимизация-это

1) улучшение программы

2) увеличение объема программы

3) увеличение количества выполняемых функций

4) уменьшение объема программы и увеличение времени ее выполнения

31. Глобальная оптимизация - это

1) оптимизация, при которой экономия времени достигается путём добавления новых циклов

2) оптимизация, при которой экономия ресурсов достигается путём изменения алгоритма всей программы или большей ее части

3) оптимизация, при которой экономия ресурсов достигается путём изменения участка программы

4) способ улучшения программ за счет уменьшения объема памяти отводимой под информационные замены

32. Чистка программы

1) в цикле имеются элементы, которые не изменяются при цикле и их выносят за пределы участка цикла

2) способ улучшения программ за счет уменьшения объема памяти отводимой под информационные замены

3) способ повышения качества за счет удаления ненужных объектов и конструкций

4) способ повышения качества программы за счет выполнения определённых вычислений на этапе трансляции

33. Время работы подпрограммы подразумевает

- 1) программу, подлежащую оптимизации следует разделить на подпрограммы и выявить процент общего времени, используемого каждой подпрограммой, количество человеко-часов для оптимизации и % отношение оптимизации подпрограммы
- 2) определение фактического времени работы каждой подпрограммы
- 3) вынесение неизменяющихся участков цикла за его пределы

34. Необходимо ли делать проверки внутри цикла

- 1) да
- 2) нет

35. Оптимизация бывает

- 1) локальная и глобальная
- 2) локальная и периодическая
- 3) глобальная и периодическая
- 4) сетевая и иерархическая

36. Локальная оптимизация - это

- 1) оптимизация, при которой экономия времени достигается путём добавления новых циклов
- 2) оптимизация, при которой экономия ресурсов достигается путём изменения участка программы
- 3) оптимизация, при которой экономия ресурсов достигается путём изменения алгоритма всей программы или большей ее части
- 4) способ улучшения программ за счет уменьшения объема памяти отводимой под информационные замены

37. Реализация действий

- 1) в цикле имеются элементы, которые не изменяются при цикле и их выносят за пределы участка цикла
- 2) способ улучшения программ за счет уменьшения объема памяти отводимой под информационные замены
- 3) способ улучшения программы за счет сокращения ее размеров: чистка, заPROCEDУРИВАНИЕ
- 4) способ повышения качества программы за счет выполнения определённых вычислений на этапе трансляции

38. Сегментация программы подразумевает

- 1) программу, подлежащую оптимизации следует разделить на подпрограммы и выявить процент общего времени, используемого каждой подпрограммой, количество человеко-часов для оптимизации и % отношение оптимизации подпрограммы
- 2) необходимость установить фактическое время работы каждой подпрограммы
- 3) вынесение неизменяющихся участков цикла за его пределы
- 4) улучшение программы за счет уменьшения объема памяти отводимой под информационные замены



39. Необходимо ли выполнять проверку вводимых пользователем данных

- 1) да
- 2) нет

40. Качество НН - это

- 1) совокупность свойств этого продукта, которые удовлетворяют определенным потребностям пользователей в соответствии с его назначением;
- 2) те свойства данного продукта, благодаря которым программный продукт может функционировать в любой программной среде;
- 3) совокупность свойств программного продукта, которые удовлетворяют требованиям ЕСНД и базовым международным стандартам.

41. Изучаемость ПН включает в себя:

- 1) удобочитаемость, тестируемость, информативность;
- 2) внедряемость, понятность, удобочитаемость;
- 3) документированность, понятность, удобочитаемость.

42. Функциональная пригодность программного продукта включает в себя:

- 1) точность, защищенность, надежность;
- 2) эффективность и внедряемость;
- 3) понятность, стабильность, надежность.

43. Понятность ПН заключается в ...

- 1) наличии в составе программы информации необходимой и достаточной для понимания назначения программы, существующих ограничений, входных и выходных данных и результатов обработки;
- 2) степени, которой пользователь может изучить назначение НН, результат ее работы и текст этой программы;
- 3) быстрой модификации с целью приспособления к изменяющимся условиям функционирования.

44. Программа является надежной, если.

- 1) выдаваемый результат работы имеет допустимые значения отклонений от аналогичных отклонений;
- 2) она продолжает свою работу при возникновении сбоев;
- 3) она при всех одинаково вводимых данных обеспечивает полную повторяемость результата.

45. Программа является эффективной, если...

- 1) она правильно работает при любых допустимых вариантах исходных данных;
- 2) объем требуемых ресурсов для ее выполнения не превышает допустимой границы;
- 3) она работает должным образом не только автономно, но и как часть информационной системы.

46. Программа является совместимой, если.

- 1) она работает должным образом не только автономно, но и как часть информационной системы;

- 2) ее качества могут быть продемонстрированы на практике;
- 3) она допускает быструю модификацию с целью приспособления к изменяющимся условиям функционирования.

47. Одним из источников ошибок в программе может быть

- 1) перевод программы;
- 2) взаимопонимание;
- 3) оба варианта правильны.

48. Сложность программы может заключаться в.

- 1) сложности в построении неформальной модели предметной области;
- 2) реализации программы, сложность в самой предметной области;
- 3) сложности в создании загрузочного модуля.

49. Что такое цикл?

- 1) Оператор выбора
- 2) Оператор условия
- 3) Оператор повторений
- 4) Оператор перехода

50. Какой из приведенных ниже циклов, является циклом с параметром?

- 1) WHILE
- 2) FOR
- 3) DO
- 4) REPEAT

51. Какой из приведенных ниже циклов, является циклом с предусловием?

- 1) CASE
- 2) FOR
- 3) REPEAT
- 4) WHILE

52. Какой из приведенных ниже циклов, является циклом с постусловием?

- 1) REPEAT
- 2) FOR
- 3) WHILE
- 4) VAR

53. Консольное приложение запускается клавишами...

- 1) F9

2) CTRL+F9

3) ALT+F9

4) F10

54. В общем виде инструкция обращения к функции выглядит так.

1) Переменная := Функция (Параметры)

2) Функция (Параметры):= Переменная

3) Переменная := Функция

4) Функция:= Функция (Параметры)

55. Какое зарезервированное слово обозначает, что переменная именно символьного типа?

1) CHR

2) CHAR

3) BOOLEAN

4) SUCC;

56. Какую структуру имеет подпрограмма?

1) Только имя подпрограммы

2) Имя и тело подпрограммы

3) Тело подпрограммы

4) Тело подпрограммы и вызов себя самой

57. При составлении блок-схемы для программы, блок-схема подпрограммы оформляется ...

1) Отдельно

2) Внутри блок-схемы программы

3) Не учитывается в основной

4) не оформляется блок-схема вообще

58. Сколько подпрограмм можно сделать в программе?

1) Только одну

2) Две

3) Сколь необходимо

4) Не больше трех

59. Для вызова подпрограммы необходимо написать...

1) Только ее имя

2) Ее имя и фактические параметры

3) Параметры подпрограммы

4) Ее имя и формальные параметры

60. В программе процедуры и функции описываются ...

1) вне основной программы

2) сразу после основной программы

3) в разделе описания подпрограмм

4) после слова begin основной программы

61. Что такое алгоритм?

1) Последовательность команд исполнителю, выполнение которых приводит к решению поставленной задачи.

2) Последовательность действий для ЭВМ.

3) Последовательность команд для ЭВМ.

4) Решение задачи

62. Способы задания алгоритма?

1) Линейный, разветвляющийся

2) Табличный, словесный, блок-схема

3) Табличный, блок-схема

4) Линейный, циклический, разветвляющийся

63. Типы алгоритма?

1) Табличный, блок-схема, словесный

2) Линейный, циклический, разветвляющийся

3) Табличный, блок-схема

4) Линейный, циклический,

64. Что такое программа?

1) Последовательность действий для решения задачи

2) Последовательность алгоритмов для решения задачи

3) Упорядоченная последовательность действий для ЭВМ, реализующая алгоритм решения кокой - либо задачи

4) Последовательность команд исполнителю, выполнение которых приводит к решению поставленной задачи.

65. Что такое вспомогательный алгоритм?

1) Набор формальных и фактических параметров

2) Процедура и функция

3) Алгоритм, помогающий при решении задачи

4) Алгоритм решения некоторой подзадачи из основной решаемой задачи

66. С помощью какого оператора в TP выводят данные на экран?

- 1) Readln
- 2) Writein
- 3) Read, Readln
- 4) Write, Writein

67. С помощью какого оператора в TP вводят данные?

- 1) Readln
- 2) Writein
- 3) Read, Readln
- 4) Write, Writein

68. Написать общий вид цикла While.

- 1) While <условие> do <параметры>
- 2) While <условие> do <оператор>
- 3) While <оператор> do <условие>
- 4) While <параметры> do <оператор>

69. Написать общий вид цикла For.

- 1) For <условие> to <операотр1> do < оператор2>
- 2) For < оператор1> to <условие> do < оператор2>
- 3) For <начальные знач. перем.> to <конечные знач. перем.> do <оператор>
- 4) For < оператор2> to <условие> do < оператор1>

70. Как выглядят операторские скобки в языке TP?

- 1) ( )
- 2) { }
- 3) (\* \*)
- 4) Begin ... End

71. Общий вид оператора IF?

- 1) If <условие> Then <оператор1> Else <оператор2>
- 2) If <оператор1> Then < оператор2> Else < условие >
- 3) If < условие > Then < оператор2> Else < оператор1>
- 4) If <параметры> Then <условие> Else <оператор>

72. Написать общий вид цикла Repeat.

- 1) Repeat <условие> Until <оператор>
- 2) Repeat <оператор1> Until <оператор2>
- 3) Repeat <оператор> Until <условие>
- 4) Repeat <параметры> Until <условие>

73. Какой раздел отсутствует в программе TP?

- 1) раздел описания переменных
- 2) раздел описания подпрограмм
- 3) раздел описания констант
- 4) раздел описания операторов

74. Как выглядят комментарии в программе TP?

- 1) ( )
- 2) { }
- 3) //
- 4) Begin .End

75. Что такое множество?

- 1) последовательность символов
- 2) совокупность однотипных элементов, рассматриваемых как единое целое
- 3) аналог таблицы, структурированный тип данных
- 4) набор символов

76. Что такое мощность множества?

- 1) количество элементов в массиве
- 2) количество элементов множества
- 3) само множество
- 4) строка в множестве

77. Как запускается программа в TP?

- 1) Alt+F5
- 2) Alt+F9
- 3) Ctrl+F9
- 4) F9

78. Раздел USES это ...

- 1) раздел описания меток
- 2) раздел описания переменных

3) раздел описания констант

4) раздел описания модулей

79. Раздел LABEL это ...

1) раздел описания меток

2) раздел описания переменных

3) раздел описания констант

4) раздел описания модулей

80. Раздел VAR это ...

1) раздел описания меток

2) раздел описания переменных

3) раздел описания констант

4) раздел описания модулей

81. Раздел CONST это ...

1) раздел описания меток

2) раздел описания переменных

3) раздел описания констант

4) раздел описания модулей

82. Что такое комментарий?

1) оператор TP

2) пояснения к программе и для программы

3) операторные скобки

4) пояснения к программе для программиста

83. Какого типа не существует в TP?

1) целочисленного

2) малочисленного

3) литерного

4) булевского

84. Сопоставьте операции над символьным типом

1. ORD(c)

a. Возвращает последующее значение символа

2. CHR(k)

b. возвращает код символа

3. PRED(c)

с. возвращает предыдущее значение символа

4 SUCC(c)

й.возвращает символ по коду

85. Какой командой подключается генератор случайных чисел?

1) Randomize

2) Graph

3) Nod

4) Dispose

86. Сопоставьте циклы

1. For

a. цикл с последующим условием

2. WHILE

b. цикл с предшествующим условием

3. REPEAT

с. цикл с параметром

87. Для чего предназначен оператор Goto?

1) оператор выбора

2) безусловный переход

3) оператор условия

4) оператор создания записи

88. Что содержит модуль CRT?

1) содержит средства управления дисплеем и клавиатурой

2) содержит пакет графических средств

3) содержатся подпрограммы обеспечивающие работу всех процедур

4) содержит средства реализации программ

89. Индивидуальность объекта - это...

1) перечень всех свойств объекта и текущие значения каждого из этих свойств

2) модель воздействия одного объекта на другие объекты в терминах изменений его состояния и передачи сообщений

3) методы и свободные подпрограммы, ассоциированные с конкретным объектом

4) характеристика объекта, которая отличает его от всех других объектов

90. Поведение объекта - это.



- 1) перечень всех свойств объекта и текущие значения каждого из этих свойств
  - 2) модель воздействия одного объекта на другие объекты в терминах изменений его состояния и передачи сообщений
  - 3) методы и свободные подпрограммы, ассоциированные с конкретным объектом
  - 4) характеристика объекта, которая отличает его от всех других объектов
1. Под ошибкой подразумевается
    - 1) место в программе, где искажение проявляется или становится очевидным
    - 2) неправильность, погрешность или неумышленное искажение объекта или процесса
    - 3) место в программе, где возникают условия для появления искажений
    - 4) исправление выявленных искажений в процессе тестирования программы
  2. Источником ошибок в программе может быть
    - 1) недостаточная квалификация специалиста
    - 2) сложность программы
    - 3) большой объем программы
    - 4) недостаточное знание заказчиком предметной области
  3. Структурный подход к разработке программы является методом борьбы с...
    - 1) переводом программы
    - 2) неквалифицированностью специалиста
    - 3) взаимопониманием
    - 4) сложностью программы
  4. Одним из признаков классификации ошибок является
    - 1) уровень сложности и устойчивости
    - 2) степень закливания
    - 3) правильность описания программы
    - 4) возможность описания программы
  5. Процесс отладки включает следующие подпроцессы:
    - 1) выявление ошибок, диагностика и локализация ошибок, исправление ошибок
    - 2) выявление ошибок и их локализация
    - 3) диагностика ошибок, исправление ошибок и повторное тестирование программы
    - 4) выявление ошибки, исправление ошибки
  6. Отладка начинается с того момента как
    - 1) не выдается сообщение об ошибках
    - 2) не выдается сообщения о синтаксических ошибках

3) программа полностью описана

4) прописаны отдельные модули программы

7. Точка обнаружения - это.

1) место в программе, где ошибка себя проявляет или становится очевидной

2) неправильность, погрешность или неумышленное искажение объекта или процесса

3) место в программе, где ошибку можно локализовать

4) место в программе, где возникают условия для появления ошибки

8. Что может являться источником ошибки в программе?

1) перевод программы

2) недостаточная квалификация специалиста

3) модульное программирование

4) объектно-ориентированное программирование

9. Контроль (проверка, испытания) программы является методом борьбы с...

1) переводом программы

2) взаимопониманием

3) сложностью программы

4) описанием программы

10. Выделяют следующие виды ошибок программ:

1) синтаксические, семантические, первичные

2) ошибки анализа, общего и физического характера

3) ошибки анализа, первичные и вторичные

4) ошибки описания, определения функций и кодирования

11. Под отладкой понимается процесс

1) нахождения и исправления ошибок

2) позволяющий получить программу, которая функционирует с требуемыми характеристиками

3) оптимизации программы

4) тиражирования программы

12. Для тестирования программы используют

1) простые тестовые данные

2) просчитанные данные

3) сложные данные

4) произвольные данные

13. Точка происхождения - это.

- 1) место в программе, где ошибка себя проявляет или становится очевидной
- 2) неправильность, погрешность или неумышленное искажение объекта или процесса
- 3) место в программе, где возникают условия для появления ошибки
- 4) место в программе, где ошибку можно локализовать

14. Выберите возможные источники ошибки в программе.

- 1) модульное программирование
- 2) трудность во взаимопонимании между заказчиком и разработчиком
- 3) сложность понимания языка программирования
- 4) объектное описание программы

15. Переход на формальные стороны взаимодействия является методом борьбы с.

- 1) переводом программы
- 2) взаимопониманием
- 3) сложностью программы
- 4) пониманием языка программирования

16. Одним из признаков классификации ошибок является

- 1) синтаксис и семантика
- 2) степень заикливания
- 3) первичные и побочные ошибки
- 4) первостепенные и второстепенные ошибки

17. Отладка бывает:

- 1) ручная и семантическая
- 2) ручная и автоматизированная
- 3) разрушающая и неразрушающая
- 4) разрушающая, семантическая, оптимизирующая

18. Тестирование - это.

- 1) оптимизация программ
- 2) действие, направленное на выявление ошибок
- 3) регистрация программы
- 4) исправление выявленных ошибок

19. Тестирование - это.

- 1) процесс создания загрузочного файла программы

- 2) запуск программы на выполнение
  - 3) процесс многократного выполнения программы с целью обнаружения максимального количества ошибок
  - 4) процесс нахождения и исправления ошибок
20. Тестовый набор данных должен включать
- 1) входные, промежуточные и выходные данные
  - 2) входные и выходные данные
  - 3) все промежуточные результаты проверки тестов и конечный результат выполнения каждой функции
  - 4) входные, выходные данные и результаты проверки каждого условия
21. Тестирование бывает
- 1) нисходящее, восходящее, промежуточное, завершенное
  - 2) структурное, функциональное, промежуточное, полное
  - 3) нисходящее, восходящее, структурное, полное
  - 4) нисходящее, восходящее, структурное, функциональное
22. Что известно при тестировании «черного ящика»?
- 1) функции программы
  - 2) внутренняя структура программы
  - 3) работа каждой функции на всей области определения
  - 4) внутренние элементы программы и связи между ними
23. При тестировании «белого ящика» исследуется...
- 1) функции программы
  - 2) внутренняя структура программы
  - 3) работа каждой функции на всей области определения
  - 4) внутренние элементы программы и связи между ними
24. К методам «белого ящика» относятся.
- 1) метод покрытия решений, метод граничных решений, метод функциональных диаграмм, метод покрытия условий
  - 2) метод эквивалентных разбиений, метод функциональных диаграмм, анализ граничных решений
  - 3) метод покрытия условий, метод покрытия операторов, метод покрытия решений, анализ граничных решений
  - 4) метод покрытия условий, метод покрытия операторов, метод покрытия решений, метод покрытия решений и условий
25. Метод эквивалентных разбиений основан на.
- 1) разработке такого числа эквивалентных тестов, достаточного для того, что бы все возможные

результаты каждого условия в решении выполнялись по крайней мере один раз

- 2) разбиении входной области программы на классы по определенным признакам
- 3) разработке достаточного количества тестов, чтобы каждое решение на этих тестах выполнялось по крайней мере один раз
- 4) выполнении каждого оператора хотя бы один раз

26. Метод покрытия условий основан на.

- 1) разработке такого числа эквивалентных тестов, достаточного для того, что бы все возможные результаты каждого условия в решении выполнялись по крайней мере один раз
- 2) разбиении входной области программы на классы по определенным признакам
- 3) разработке достаточного количества тестов, чтобы возможные результаты каждого условия в решении выполнялось по крайней мере один раз
- 4) выполнении каждого оператора хотя бы один раз

27. Тестирование «черного ящика» выполняется

- 1) на ранних этапах разработки программы
- 2) когда разработан весь программный продукт и протестированы отдельные его модули
- 3) на поздних стадиях тестирования программы
- 4) на ранних стадиях тестирования программы

28. Техника «черного ящика» ориентирована на...

- 1) выявление класса ошибок
- 2) выявление отдельных ошибок
- 3) сокращение количества тестовых вариантов
- 4) увеличение количества тестовых наборов

29. Тестирование включает в себя ...

- 1) создание текстового, загрузочного файла и их проверка
- 2) разработка тестов и непосредственное тестирование по ним
- 3) проверка разработанного набора тестов на исполняемом файле
- 4) составление алгоритма решения задачи, текста программы, набора тестовых данных и их проверка

30. Чему равна вероятность наличия необнаруженных ошибок в какой-то части программы?

- 1) обратно пропорциональна числу ошибок обнаруженных в программе
- 2) количеству обнаруженных в программе ошибок
- 3) пропорциональна числу ошибок обнаруженных в программе
- 4)  $1/3$  числу обнаруженных ошибок

31. Что известно при тестировании «белого ящика»?

- 1) функции программы
- 2) внутренняя структура программы
- 3) работа каждой функции на всей области определения
- 4) внутренние элементы программы и связи между ними

32. При тестировании «черного ящика» исследуется.

- 1) функции программы
- 2) внутренняя структура программы
- 3) работа каждой функции на всей области определения
- 4) внутренние элементы программы и связи между ними

33. К методам «черного ящика» относятся...

- 1) метод покрытия решений, метод граничных решений, метод функциональных диаграмм, метод покрытия условий
- 2) метод эквивалентных разбиений, метод функциональных диаграмм, анализ граничных решений
- 3) метод покрытия условий, метод покрытия операторов, метод покрытия решений, анализ граничных решений
- 4) метод покрытия условий, метод покрытия операторов, метод покрытия решений, метод покрытия решений и условий

34. Метод покрытия операторов при тестировании программ основан на.

- 1) разработке такого числа эквивалентных тестов, достаточного для того, что бы все возможные результаты каждого условия в решении выполнялись по крайней мере один раз
- 2) разбиении входной области программы на классы по определенным признакам
- 3) разработке достаточного количества тестов, чтобы каждое решение на этих тестах выполнялось по крайней мере один раз
- 4) выполнении каждого оператора хотя бы один раз

35. Граничные условия - это

- 1) условия, ситуация, возникающая непосредственно на границе выше или ниже границ входных или выходных элементов класса эквивалентности
- 2) ситуация, возникающая непосредственно на промежуточных элементах класса эквивалентности
- 3) условия, ситуация, возникающие внутри программы, когда выполнены все тестовые наборы

36. При тестировании программ методами «черного ящика» необходимо разрабатывать набор тестов, который.

- 1) показывает нормальное функционирование программы
- 2) выявляет все ошибки программы и по ним позволяет оптимизировать программу
- 3) показывает нормальное и аномальное функционирование программы

37. Тестирование «белого ящика» выполняется

- 1) на ранних этапах разработки программы
  - 2) когда разработан весь программный продукт и протестированы отдельные его модули
  - 3) на поздних стадиях тестирования программы
  - 4) на ранних стадиях тестирования программы
38. Тестирование «черного ящика» обеспечивает поиск следующих категорий ошибок:
- 1) ошибок во внутренних структурах данных
  - 2) ошибок интерфейса
  - 3) ошибок во внешних структурах данных
  - 4) ошибок в циклах и ветвлениях
  - 5) ошибок характеристик
39. К программным средствам защиты программного продукта не относят....
- 1) криптографическую защиту
  - 2) ограничение доступа к программному продукту
  - 3) патентную защиту
  - 4) нестандартное форматирование диска, на котором находится программный продукт
40. Лицензирование программного продукта относится к.
- 1) правовой защите ПН
  - 2) программной защите НН
  - 3) технической защите ПН
  - 4) физической защите НН
41. Каким знаком обозначается авторское право на программный продукт?
- 1) ©
  - 2) тм
  - 3) ®
42. Каким знаком обозначается регистрация права на программный продукт?
- 1) ©
  - 2) тм
  - 3) ®
43. Этап Эволюции при сопровождении программного продукта предполагает.
- 1) выявление и устранение обнаруженных ошибок, тиражирование, контроль за распространением версии, введение новых функций программы и т.д
  - 2) внесение изменения в программу в ответ на изменившиеся условия
  - 3) использование всех возможных и невозможных способов для поддержания жизни в старой и

распадающейся на части программной системе

4) проектирование программного продукта, тестирование, тиражирование и утилизацию

44. Этап Сохранение при сопровождении программного продукта предполагает...

1) выявление и устранение обнаруженных ошибок, тиражирование, контроль за распространением версии, введение новых функций программы и т.д

2) внесение изменения в программу в ответ на изменившиеся условия

3) использование всех возможных и невозможных способов для поддержания жизни в старой и распадающейся на части программной системе

4) проектирование программного продукта, тестирование, тиражирование и утилизацию

45. Этап Чистое сопровождение при сопровождении программного продукта предполагает.

1) выявление и устранение обнаруженных ошибок, тиражирование, контроль за распространением версии, введение новых функций программы и т.д

2) внесение изменения в программу в ответ на изменившиеся условия

3) использование всех возможных и невозможных способов для поддержания жизни в старой и распадающейся на части программной системе

4) проектирование программного продукта, тестирование, тиражирование и утилизацию

1. Существует две основные модели организации коллектива при разработке ПО:

1) иерархическая модель и модель группы

2) структурная и объектная модель

3) иерархическая и объектная модель

4) модель группы и сетевая модель

2. Какая модель коллективной разработки программного продукта определяет структуру коллектива с точки зрения отдела кадров?

1) модель группы

2) иерархическая модель

3) структурная модель

4) сетевая модель

3. Какая модель коллективной разработки программного продукта не определяет структуру коллектива с точки зрения отдела кадров?

1) модель группы

2) иерархическая модель

3) структурная модель

4) сетевая модель

4. Основными недостатками иерархической модели коллективной разработки программных продуктов является:



- 1) несогласованное представление о разных сторонах проекта
  - 2) нехватка информации
  - 3) разрозненная связь с внешними источниками информации
  - 4) сложность расстановки приоритетов
5. Основными недостатками групповой модели коллективной разработки программных продуктов является:
- 1) несогласованное представление о разных сторонах проекта
  - 2) нехватка информации
  - 3) разрозненная связь с внешними источниками информации
  - 4) сложность расстановки приоритетов
6. Для скрытия недостатков иерархической модели коллективной разработки программных продуктов предусматривают
- 1) сплочение коллектива путем приобретения большого числа заказов
  - 2) распределение обязанностей руководителя между членами коллектива
  - 3) объединение обязанностей руководителя и отдельных членов коллектива
  - 4) определение целей проекта и распределение обязанностей в соответствии с целями
7. В чем заключается задача модели проектной группы при коллективной разработке программного продукта?
- 1) сплочение коллектива путем приобретения большого числа заказов
  - 2) распределение обязанностей руководителя между членами коллектива
  - 3) объединение обязанностей руководителя и отдельных членов коллектива
  - 4) определение целей проекта и распределение обязанностей между членами группы
8. Основной целью менеджера продукта при коллективной разработке программного продукта является:
- 1) удовлетворение требований заказчика
  - 2) соблюдение ограничений проекта
  - 3) соответствие спецификациям
  - 4) выпуск программного продукта только после выявления и устранения проблем
9. Основной целью менеджера программы при коллективной разработке программного продукта является:
- 1) удовлетворение требований заказчика
  - 2) соблюдение ограничений проекта
  - 3) соответствие спецификациям
  - 4) выпуск только после выявления и устранения проблем
10. Основной целью разработчика при коллективной разработке программного продукта является:

1) удовлетворение требований заказчика

2) соблюдение ограничений проекта

3) соответствие спецификациям

4) выпуск программного продукта только после выявления и устранения проблем

11. Основной целью тестера при коллективной разработке программного продукта является:

1) удовлетворение требований заказчика

2) соблюдение ограничений проекта

3) соответствие спецификациям

4) выпуск программного продукта только после выявления и устранения проблем

12. Основной целью инструктора при коллективной разработке программного продукта является:

1) удовлетворение требований заказчика

2) соблюдение ограничений проекта

3) повышение эффективности труда пользователя

4) выпуск программного продукта только после выявления и устранения проблем

13. Основной целью инструктора при коллективной разработке программного продукта является:

1) удовлетворение требований заказчика

2) простота развертывания и постоянное сопровождение программного продукта

3) повышение эффективности труда пользователя

4) выпуск программного продукта только после выявления и устранения проблем

14. Главная задача менеджера продукта при коллективной разработке программного продукта заключается...

1) в формировании общего представления о поставленной задаче и о том, как ее решать

2) в ведении процесса разработки с учетом всех ограничений

3) в испытании продукта в реальных условиях

4) в повышении эффективности труда пользователей

14. Главная задача тестера при коллективной разработке программного продукта заключается...

1) в формировании общего представления о поставленной задаче и о том, как ее решать

2) в ведении процесса разработки с учетом всех ограничений

3) в испытании продукта в реальных условиях

4) в повышении эффективности труда пользователей

15. Главная задача менеджера продукта при коллективной разработке программного продукта заключается...

1) в формировании общего представления о поставленной задаче и о том, как ее решать

2) в ведении процесса разработки с учетом всех ограничений

3) в испытании продукта в реальных условиях

4) в повышении эффективности труда пользователей

16. Главная задача инструктора при коллективной разработке программного продукта состоит...

1) в формировании общего представления о поставленной задаче и о том, как ее решать

2) в ведении процесса разработки с учетом всех ограничений

3) в испытании продукта в реальных условиях

4) в повышении эффективности труда пользователей

17. Главная задача логистика при коллективной разработке программного продукта состоит...

1) в формировании общего представления о поставленной задаче и о том, как ее решать

2) в проверке, чтобы все серверы развертывания и рабочие станции пользователей удовлетворяли указанным требованиям

3) в испытании продукта в реальных условиях

4) в повышении эффективности труда пользователей

18. Кто из членов группы при коллективной разработке программных продуктов составляет график работ?

1) менеджер продукта

2) менеджер программы

3) логистик

4) инструктор

19. Кто из членов группы при коллективной разработке программных продуктов выполняет проектирование архитектуры программного продукта?

1) менеджер продукта

2) разработчик

3) логистик

4) инструктор

20. Кто из членов группы при коллективной разработке программных продуктов разрабатывает стратегию, планы, графики и сценарии тестирования?

2) разработчик

1) менеджер продукта

3) логистик

4) тестер

21. Кто из членов группы при коллективной разработке программных продуктов составляет документацию, определяет требования к резервному копированию данных и разрабатывает план восстановления на случай отказа систем?

1) менеджер продукта

2) разработчик

3) логистик

4) тестер

22. Кто из членов группы при коллективной разработке программных продуктов участвует в создании пользовательского интерфейса, сокращая тем самым затраты на сопровождение продукта и поддержку пользователей?

1) менеджер продукта

2) разработчик

3) инструктор

4) тестер

23. К недостатками иерархической модели коллективной разработки программных продуктов можно отнести:

1) несогласованное представление о разных сторонах проекта

2) невозможностью учесть все особенности проекта

3) разрозненная связь с внешними источниками информации

4) отсутствием полноценной связи между всеми участниками проекта, так как вся информация идет в одном направлении — вверх по иерархии, к главному менеджеру

24. К недостаткам иерархической модели коллективной разработки программных продуктов можно отнести:

1) несогласованность личных планов членов группы

2) невозможностью учесть все особенности проекта

3) отсутствие опыта, снижающее эффективность коллективной работы

4) отсутствием полноценной связи между всеми участниками проекта, так как вся информация идет в одном направлении — вверх по иерархии, к главному менеджеру

25. Какие задачи необходимо решить, чтобы проект считался удачным?

1) удовлетворить требования заказчика

2) соблюсти ограничения

3) спроектировать систему по объектно-ориентированному методу

4) выполнить спецификации, основанные на требованиях пользователей

- 5) выпустить продукт только после выявления и устранения всех проблем
- 6) выполнить программный продукт с учетом ситуации на рынке программ
- 7) гарантировать простоту развертывания и управления

### 3.1.3 Материал для составления экзаменационных билетов

#### Перечень экзаменационных теоретических вопросов

##### Введение

1. Дать определения понятий: программа, программное обеспечение, задача, приложение.
2. Дать определения понятий: предметная область, постановка задачи, алгоритм решения задачи.
3. Основные свойства алгоритмов.

##### Основная категория специалистов

##### занятых разработкой и эксплуатацией программ

1. Уровень категории специалистов охваченных процессом подготовки программ.
2. Охарактеризовать деятельность системного программиста, прикладного программиста.
3. Охарактеризовать деятельность программиста аналитика, постановщика задач, администратора баз данных.
4. Описать работу конечного пользователя.
5. Охарактеризовать взаимодействие специалистов связанных с созданием и эксплуатацией программ.

##### Характеристика программного продукта

1. Дать определения понятий: утилитарные программы, программные продукты. Программный продукт.
2. Этапы создания ПП, сопровождения ПП.
3. Дать определение основных характеристик программ.
4. Дать определение жизненного цикла программного продукта

##### Защита программных продуктов

1. Методы защиты программных продуктов.
2. Цели защиты программных продуктов.
3. Программные системы защиты от несанкционированного доступа.
4. Правовые методы защиты программ.
5. Патентная защита лицензионные соглашения. Типы лицензий на ПП.
6. В чем состоит авторское право разработчика программы.

##### Классификация программных продуктов

1. Как можно классифицировать программные продукты.
2. Как можно классифицировать системное программное обеспечение. Структура СПО.
3. Что входит в системное программное обеспечение.
4. Дать определение операционной системы. Рейтинг ОС, классификация, производительность
5. Деление ОС для персонального компьютера. Приведите примеры.
6. Дать основные характеристики ОС. Операционные оболочки.

#### Сервисное программное обеспечение

1. Дать определение базового программного обеспечения компьютера.
2. Что входит в набор сервисных (дополнительно устанавливаемых) программ.
3. Дать определение программы утилиты. Какие комплекты утилит вы знаете. Приведите пример.
4. Какие антивирусные продукты вы знаете. Каковы критерии их оценки.
5. Что входит в минимальный состав системного программного обеспечения.

#### Инструментарий технологии программирования

1. Состав и назначение инструментария технологии программирования.
2. Что такое программотехника и ее инструментарий.
3. Назовите виды инструментальных средств, для разработки программных продуктов и дайте им краткую характеристику.
4. Дайте определение CASE-технологии. Когда она используется.
5. Какие виды языков программирования вы знаете. Классификация языков программирования.
6. Что входит в состав системы программирования.
7. Расскажите об инструментальной среде конечного пользователя.

#### Интегрированные среды разработки программ

1. Приведите примеры приемов надежного программирования.
2. Какие методы повышения производительности труда программиста вы знаете.
3. Производительность труда программиста и CASE - технология.
4. Приведите примеры и краткую характеристику программных продуктов для создания приложений.

#### Пакеты прикладных программ

1. Дайте характеристику пакетов прикладных программ и их классификацию.
2. Расскажите об проблемно-ориентированных ППП. Основные тенденции в области развития проблемно-ориентированных программных средств.
3. Расскажите об ППП автоматизированного проектирования. Приведите примеры.
4. Расскажите об ППП общего назначения. Какие ППП можно считать офисными.
5. Расскажите об методо-ориентированных ППП их характеристика. Приведите примеры.
6. Расскажите об коммуникационных ППП. Использование программных средств мультимедиа.

7. Дайте характеристику основных компонентов системы искусственного интеллекта.

#### Стиль программирования

1. Выбор и обоснование языка программирования на примерах.
2. Приведите примеры использования библиотек программ, встроенных функций.
3. Охарактеризуйте методы структурирования программ.
4. Дайте определение и характеристику объектно-ориентированного программирования.
5. По каким признакам происходит оценка эффективности программ.
6. Перечислите основные этапы отладки и сопровождения программных продуктов. Приведите примеры.
7. Охарактеризуйте ошибки программного обеспечения основные причины, источники и классификацию.

### Перечень экзаменационных практических задач

1. Создать проект приложения с регулятором прозрачности. В качестве источника значений использовать компонент QScrollBar.
2. Создать проект приложения с регулятором прозрачности. В качестве источника значений использовать компонент QSlider.
3. Создать проект простого вычислителя, выполняющего операцию сложения. В качестве операндов использовать действительные числа.
4. Создать проект простого вычислителя, выполняющего операцию вычитания. В качестве операндов использовать действительные числа.
5. Создать проект простого вычислителя, выполняющего операцию деления. В качестве операндов использовать действительные числа.
6. Создать проект простого вычислителя, выполняющего операцию умножения. В качестве операндов использовать действительные числа.
7. Используя диалог открытия файла, загрузить текстовый файл с расширением txt в поле QPlainTextEdit. Использовать фильтр на расширение.
8. Используя диалог открытия файла, загрузить текстовый файл с расширением html в поле QTextEdit. Использовать фильтр на расширение.
9. Используя диалог сохранения файла, сохранить содержимое поля QPlainTextEdit в текстовый файл с расширением txt. Использовать фильтр на расширение и его автоматическую подстановку.
10. Используя диалог сохранения файла, сохранить содержимое поля QTextEdit в текстовый файл с расширением html. Использовать фильтр на расширение и его автоматическую подстановку.
11. Используя диалог выбора цвета, изменить цвет клиентской области формы по нажатию клавиши «Цвет».
12. Используя диалог выбора цвета, изменить цвет шрифта произвольно названного компонента QLabel по нажатию клавиши «Цвет текста».
13. Создать главное меню программы следующей структуры:
  - Файл (Открыть, Сохранить, Выход) Назначить соответствующие горячие клавиши.
14. Создать главное меню программы следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
  - Правка (Вырезать, Копировать, Вставить) Назначить соответствующие горячие клавиши.
15. Создать главное меню программы следующей структуры:
  - Файл (Открыть, Сохранить, Выход)
 По нажатию пунктов «Открыть» и «Сохранить» вызвать соответствующие действия для произвольно названного поля QPlainTextEdit.
16. В модуле Unit1 пустого проекта создать функцию вычисляющую выражение  $f(x) = e^x \cdot \sin x - e^{-x} \cdot \cos x$   
Использовать прототип: **float f(float x);**
17. В модуле Unit1.pas пустого проекта создать функцию вычисляющую выражение  $f(x) = e^{\sin x} - \sin e^x$   
Использовать прототип: **float f(float x);**
18. Создать модуль UnitX. Включить в него функцию, вычисляющую выражение  $f(x) = e^{\sin x}$   
Использовать прототип: **float f(float x);**
19. Создать модуль plus\_minus. Включить в него две функции (сложения и вычитания) с использованием следующих прототипов:  
**float plus(float a, float b);**  
**float minus(float a, float b);**
20. Подключить модуль UnitX, содержащий функцию f, и использовать в вычислениях. Использовать прототип: **float f(int x);**
21. Подключить модуль Div\_Mod, содержащий две функции ndiv и nmod, и использовать в вычислениях.



Использовать прототипы:

```
int ndiv(int a,int b);
```

```
int nmod(int a,int b);
```

22. Создать библиотеку динамической компоновки с именем Proj.dll, содержащую в секции экспорта функцию proсх следующего прототипа

```
void proсх(float a,float b,float& c,float& d);
```

В результате выполнения процедуры параметры с и d принимают значения суммы и разности параметров а и b соответственно.

23. Создать библиотеку динамической компоновки с именем Project1.dll, содержащую в секции экспорта функцию f следующего прототипа

```
float f(float x);
```

Эта функция должна вычислять выражение  $f(x) = 410 \cdot \sin x$ .

24. Подключить библиотеку динамической компоновки с именем Proj.dll, содержащую в секции экспорта процедуру proсх следующего прототипа

```
void proсх(float a,float b,float& c,float& d);
```

Использовать статический импорт.

25. Подключить библиотеку динамической компоновки с именем Project1.dll, содержащую в секции экспорта функцию f следующего прототипа

```
float f(float x);
```

Использовать статический импорт.

26. Подключить библиотеку динамической компоновки с именем Proj.dll, содержащую в секции экспорта процедуру proсх следующего прототипа

```
void proсх(float a,float b,float& c,float& d);
```

Использовать динамический импорт.

27. Подключить библиотеку динамической компоновки с именем Project1.dll, содержащую в секции экспорта функцию f следующего прототипа

```
float f(float x);
```

Использовать динамический импорт.

## 3.2. Задания для оценки освоения МДК 03.02 Инструментальные средства разработки программного обеспечения

### 3.2.1 Типовой тест к дифференцированному зачету

1. Программное средство, предназначенное для поддержки разработки других программ, называется -.
  - 1) аппаратным инструментом
  - 2) программным инструментом
  - 3) программной средой
  - 4) инструментарий технологии программирования
2. Анализаторы обеспечивают.
  - 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
  - 2) автоматически приводить документы к другой форме представления или переводить документ одного вида к документу другого вида
  - 3) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ
  - 4) выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода
3. Преобразователи обеспечивают.
  - 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
  - 2) автоматически приводить документы к другой форме представления или переводить документ одного вида к документу другого вида
  - 3) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ
  - 4) выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода
4. Сколько классов инструментальных средств выделяют в инструментальной среде разработки и сопровождения программ?
  - 1) 2
  - 2) 4
  - 3) 3
  - 4) 5
5. Среда программирования предназначена для.
  - 1) конструирования тех или иных программных продуктов и документов на различных этапах жизненного цикла
  - 2) автоматического перевода документов к другой форме представления или перевода документа одного вида к документу другого вида
  - 3) поддержки ранних этапов разработки программ и автоматической генерации программ по спецификациям
  - 4) поддержки процессов программирования (кодирования), тестирования и отладки программ
6. Инструментальные среды программирования бывают
  - 1) языково-ориентированные среды и среды общего назначения
  - 2) объектно-ориентированные и языково-ориентированные среды
  - 3) среды общего назначения и прикладные среды
  - 4) среды общего назначения, прикладные среды, логические и математические среды
7. Для поддержки разработки программного продукта на каком-либо одном языке программирования используют.
  - 1) среду программирования общего назначения
  - 2) языково-ориентированную среду программирования
  - 3) интерпретирующую среду программирования

- 4) прикладную среду программирования
8. Синтаксически-управляемая инструментальная среда программирования базируется на знании
  - 1) семантики языка программирования
  - 2) синтаксиса языка программирования
  - 3) синтаксиса и семантики языка программирования
  - 4) основных управляющих структур языка программирования
9. На рисунке представлена классификация
  - 1) инструментальной системы технологии программирования
  - 2) инструментальной среды разработки и сопровождения программ
  - 3) рабочего места компьютерной технологии
  - 4) языков программирования
10. Инструментальная система технологии программирования - это...
  - 1) программное средство, предназначенное для поддержки разработки других программ
  - 2) устройство компьютера, специально предназначенное для поддержки разработки программного средства
  - 3) интегрированная совокупность программных и аппаратных инструментов, поддерживающая все процессы разработки и сопровождения больших программных продуктов
  - 4) логически связанная совокупность программных и аппаратных инструментов поддерживающих разработку ИИ
11. Устройство компьютера, специально предназначенное для поддержки разработки программного средства, называется -.
  - 1) аппаратным инструментом
  - 2) программным инструментом
  - 3) программной средой
  - 4) инструментарий технологии программирования
12. Редакторы обеспечивают.
  - 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
  - 2) автоматически приводить документы к другой форме представления или переводить документ одного вида к документу другого вида
  - 3) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ
  - 4) выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода
13. Инструменты, поддерживающие процесс выполнения программ обеспечивают.
  - 1) конструирование тех или иных программных продуктов и документов на различных этапах жизненного цикла
  - 2) автоматический привод документов к другой форме представления или перевод документа одного вида к документу другого вида
  - 3) возможность выполнять на компьютере описание процессов или отдельных их частей, представленных в виде, отличном от машинного кода
  - 4) статическую обработку документов, осуществляя различные виды их контроля, выявление определенных их свойств и накопление статистических данных, либо динамический анализ программ
14. Инструментальная система технологии программирования предназначена для.
  - 1) поддержки всех процессов разработки и сопровождения в течение всего жизненного цикла ИС и ориентирована на коллективную разработку больших программных систем с длительным жизненным циклом
  - 2) автоматического перевода документов к другой форме представления или перевода документа одного вида к документу другого вида
  - 3) поддержки ранних этапов разработки программ и автоматической генерации программ по спецификациям

- 4) поддержки процессов программирования (кодирования), тестирования и отладки программ
15. Рабочее место компьютерной технологии предназначено для.
- 1) конструирования тех или иных программных продуктов и документов на различных этапах жизненного цикла
  - 2) автоматического перевода документов к другой форме представления или перевода документа одного вида к документу другого вида
  - 3) поддержки ранних этапов разработки программ и автоматической генерации программ по спецификациям
  - 4) поддержки процессов программирования (кодирования), тестирования и отладки программ
16. Инструментальные среды программирования содержат
- 1) редактор, анализатор и компилятор
  - 2) редактор, интерпретатор и компилятор
  - 3) интерпретатор, компилятор, преобразователь
  - 4) редактор и интерпретатор
17. Для поддержки разработки программного продукта на разных языках программирования (например, текстовый редактор, редактор связей или интерпретатор языка целевого компьютера) используют.
- 1) среду программирования общего назначения
  - 2) языково-ориентированную среду программирования
  - 3) интерпретирующую среду программирования
  - 4) прикладную среду программирования
18. На рисунке представлена классификация
- 1) инструментальной системы технологии программирования
  - 2) инструментальной среды программирования
  - 3) рабочего места компьютерной технологии
  - 4) языков программирования
19. При использовании компьютерных технологий для разработки ПН жизненный цикл ПН представлен следующей цепочкой:
- 1) прототипирование - кодогенерация - комплексная отладка и тестирование - аттестация, применение, сопровождение
  - 2) прототипирование - разработка спецификаций - автоматизированный контроль спецификаций - кодогенерация - комплексная отладка и тестирование - аттестация, применение, сопровождение
  - 3) разработка спецификаций - автоматизированный контроль спецификаций - кодогенерация - комплексная отладка и тестирование - аттестация, применение, сопровождение
  - 4) прототипирование - разработка спецификаций - кодогенерация - аттестация, применение, сопровождение
20. Основными чертами инструментальной системы технологии программирования являются.
- 1) массовость, дискретность, результативность, определенность, понятность
  - 2) комплексность, ориентированность на коллективную разработку, технологическая определенность, интегрированность
  - 3) актуальность, непротиворечивость, полнота
  - 4) комплексность, актуальность, интегрированность, массовость, понятность
21. Современные крупные проекты информационных систем характеризуются следующими особенностями:
- 1) сложность описания, требующая тщательного моделирования и анализа данных и процессов
  - 2) наличие совокупности тесно взаимодействующих компонентов
  - 3) наличие прямых аналогов, ограничивающее возможность использования каких-либо типовых проектных решений
  - 4) невозможность интеграции существующих и вновь разрабатываемых приложений;
22. Под CASE-средства понимаются программные средства, поддерживающие.
- 1) процессы создания и сопровождения ИС, включая анализ и формулировку требований, проектирование прикладного ПО
  - 2) процессы тиражирования программного продукта

- 3) процессы создания и эксплуатации программного продукта
  - 4) процессы компилирования и интерпретации программных продуктов
23. Репозиторий Case - средства - это.
- 1) специализированная база данных проекта, предназначенная для отображения состояния проектируемой системы в каждый момент времени
  - 2) компонент, обеспечивающий создание и редактирование в интерактивном режиме элементов диаграмм и связей между ними
  - 3) компонент, служащий для контроля правильности построения диаграмм в заданной методологии проектирования
  - 4) компонент, позволяющий получать информацию о проектах в виде отчетов
  - 5) компонент, выполняющий запуск проекта, задание начальных параметров и назначение и изменение прав доступа к элементам проекта
  - 6) набор системных утилит по обслуживанию репозитория
24. Графический редактор Case - средства - это.
- 1) компонент, обеспечивающий создание и редактирование в интерактивном режиме элементов диаграмм и связей между ними
  - 2) компонент, служащий для контроля правильности построения диаграмм в заданной методологии проектирования
  - 3) компонент, позволяющий получать информацию о проектах в виде отчетов
  - 4) компонент, выполняющий запуск проекта, задание начальных параметров и назначение и изменение прав доступа к элементам проекта
25. Верификатор Case - средства - это.
- 1) компонент, служащий для контроля правильности построения диаграмм в заданной методологии проектирования
  - 2) компонент, позволяющий получать информацию о проектах в виде отчетов
  - 3) компонент, выполняющий запуск проекта, задание начальных параметров и назначение и изменение прав доступа к элементам проекта
  - 4) набор системных утилит по обслуживанию репозитория
26. Документатор проекта Case - средства - это.
- 1) компонент, позволяющий получать информацию о проектах в виде отчетов
  - 2) компонент, выполняющий запуск проекта, задание начальных параметров и назначение и изменение прав доступа к элементам проекта
  - 3) набор системных утилит по обслуживанию репозитория
  - 4) компонент, обеспечивающий создание и редактирование в интерактивном режиме элементов диаграмм и связей между ними
27. Сервис Case - средства - это.
- 1) компонент, служащий для контроля правильности построения диаграмм в заданной методологии проектирования
  - 2) компонент, позволяющий получать информацию о проектах в виде отчетов
  - 3) компонент, выполняющий запуск проекта, задание начальных параметров и назначение и изменение прав доступа к элементам проекта
  - 4) набор системных утилит по обслуживанию репозитория
28. Администратор проекта Case - средства - это.
- 1) компонент, служащий для контроля правильности построения диаграмм в заданной методологии проектирования
  - 2) компонент, позволяющий получать информацию о проектах в виде отчетов
  - 3) компонент, выполняющий запуск проекта, задание начальных параметров и назначение и изменение прав доступа к элементам проекта
  - 4) набор системных утилит по обслуживанию репозитория
29. Какие методологии проектирования используют Case - средства?
- 1) структурного и модульного проектирования
  - 2) структурного и объектно-ориентированного проектирования

- 3) объектно-ориентированного и нисходящего проектирования
- 4) нисходящего и восходящего проектирования
30. Структурное проектирование системы основано на.
  - 1) объектно-ориентированной декомпозиции
  - 2) алгоритмической декомпозиции
  - 3) модульной декомпозиции
  - 4) функциональной декомпозиции
31. Объектно-ориентированное проектирование системы основано на.
  - 1) объектно-ориентированной декомпозиции
  - 2) алгоритмической декомпозиции
  - 3) модульной декомпозиции
  - 4) функциональной декомпозиции
32. Case - средства представляют собой.
  - 1) набор инструментальных средств для проектирования программного продукта
  - 2) набор программных средств для сопровождения программного продукта
  - 3) набор программных и инструментальных средств, поддерживающие все процессы жизненного цикла программного продукта
  - 4) набор аппаратных средств, поддерживающих все процессы жизненного цикла программного продукта
33. Компания-разработчик приобрела новое Case - средство. Сразу ли компания получит ожидаемый результат от применения новой технологии?
  - 1) да
  - 2) нет
34. Сколько классов Case - средств выделяют?
  - 1) 5
  - 2) 3
  - 3) 7
  - 4) 2
35. Case - средства анализа и проектирования, предназначенные для
  - 1) моделирования данных и генерации схем баз данных
  - 2) построения и анализа моделей деятельности организаций (предметной области) или моделей проектируемой системы
  - 3) обеспечения комплексной поддержки требований к создаваемой системе
  - 4) поддержки всего жизненного цикла программного продукта
36. Case - средства управления требованиями предназначены для
  - 1) моделирования данных и генерации схем баз данных
  - 2) построения и анализа моделей деятельности организаций (предметной области) или моделей проектируемой системы
  - 3) обеспечения комплексной поддержки требований к создаваемой системе
  - 4) поддержки всего жизненного цикла программного продукта
37. Case - средства проектирования баз данных предназначены для
  - 1) моделирования данных и генерации схем баз данных
  - 2) построения и анализа моделей деятельности организаций (предметной области) или моделей проектируемой системы
  - 3) обеспечения комплексной поддержки требований к создаваемой системе
  - 4) поддержки всего жизненного цикла программного продукта
38. На каких стандартах базируется технология освоения и внедрения Case - средств?
  - 1) ГОСТ 2107-99
  - 2) IEEE (IEEE Std 1348-1995 и IEEE Std 1209-1992)
  - 3) AES
  - 4) ISO
39. Из каких этапов состоит процесс освоения и внедрения Case - средств?

- 1) определение потребностей в CASE-средствах, оценка и выбор CASE-средств, практическое внедрение CASE-средств
  - 2) определение потребностей в CASE-средствах, оценка и выбор CASE-средств, выполнение пилотного проекта, практическое внедрение CASE-средств
  - 3) определение потребностей в CASE-средствах, проектирования CASE-средств, практическое применение CASE-средств
  - 4) проектирование CASE-средств, оценка и внедрение CASE-средств, практическое применение CASE-средств
40. Критериями для выбора CASE-средств могут являться
- 1) открытая архитектура, поддержка полного жизненного цикла ИС с обеспечением эволюционности ее развития, обеспечение целостности проекта, независимость от программно-аппаратной платформы и СУБД
  - 2) модифицируемость, простота, эффективность, учет человеческого фактора, многоплатформенность
  - 3) закрытая архитектура, поддержка полного жизненного цикла ИС с обеспечением эволюционности ее развития, простота, эффективность
  - 4) максимальная зависимость от программных и аппаратных средств системы и характеристик самой системы, жесткая привязка к конкретным информационным процессам, прочность внутренней связи отдельных компонентов системы
41. Комплексность компьютерной поддержки разработки ИЛ с использованием инструментальной системы технологии программирования означает
- 1) что система технологии программирования охватывает все процессы разработки и сопровождения ИС и что продукция этих процессов согласована и взаимоувязана
  - 2) что система технологии программирования должна поддерживать управление работой коллектива и для разных членов этого коллектива обеспечивать разные права доступа к различным фрагментам продукции технологических процессов
  - 3) что все инструменты объединены единым пользовательским интерфейсом
  - 4) что инструменты действуют в соответствии с фиксированной информационной схемой системы, определяющей зависимость различных используемых в системе фрагментов данных друг от друга
42. Ориентированность инструментальной системы технологии программирования на коллективную разработку означает
- 1) что система технологии программирования охватывает все процессы разработки и сопровождения ИС и что продукция этих процессов согласована и взаимоувязана
  - 2) что система технологии программирования должна поддерживать управление работой коллектива и для разных членов этого коллектива обеспечивать разные права доступа к различным фрагментам продукции технологических процессов
  - 3) что все инструменты объединены единым пользовательским интерфейсом
  - 4) что инструменты действуют в соответствии с фиксированной информационной схемой системы, определяющей зависимость различных используемых в системе фрагментов данных друг от друга
43. Технологическая определенность инструментальной системы технологии программирования означает
- 1) что система технологии программирования охватывает все процессы разработки и сопровождения ИС и что продукция этих процессов согласована и взаимоувязана
  - 2) что система технологии программирования должна поддерживать управление работой коллектива и для разных членов этого коллектива обеспечивать разные права доступа к различным фрагментам продукции технологических процессов
  - 3) что ее комплексность ограничивается рамками какой-либо конкретной технологии программирования
  - 4) что инструменты действуют в соответствии с фиксированной информационной схемой системы, определяющей зависимость различных используемых в системе фрагментов данных друг от друга
44. Интегрированность инструментальной системы технологии программирования по данным означает
- 1) что система технологии программирования охватывает все процессы разработки и сопровождения ИС и что продукция этих процессов согласована и взаимоувязана
  - 2) что система технологии программирования должна поддерживать управление работой коллектива и

для разных членов этого коллектива обеспечивать разные права доступа к различным фрагментам продукции технологических процессов

3) что ее комплексность ограничивается рамками какой-либо конкретной технологии программирования

4) что инструменты действуют в соответствии с фиксированной информационной схемой системы, определяющей зависимость различных используемых в системе фрагментов данных друг от друга

45. Интегрированность инструментальной системы технологии программирования по пользовательскому интерфейсу означает

1) что система технологии программирования охватывает все процессы разработки и сопровождения ПС и что продукция этих процессов согласована и взаимоувязана

2) что система технологии программирования должна поддерживать управление работой коллектива и для разных членов этого коллектива обеспечивать разные права доступа к различным фрагментам продукции технологических процессов

3) что ее комплексность ограничивается рамками какой-либо конкретной технологии программирования

4) что все инструменты объединены единым пользовательским интерфейсом



### 3.2.2 Вопросы для проведения экзамена

1. Опишите понятие псевдокода.
2. Опишите понятие ошибки в программном средстве.
3. Опишите инжиниринг ПО в инструментальной среде Flash Professional на языке ActionScript.
4. Опишите понятие программного средства.
5. Что такое инвариант цикла?
6. Опишите инжиниринг ПО в инструментальной среде IDE Net Beans на языке Java.
7. Опишите понятие надежности программного средства.
8. Опишите понятие визуальных сред программирования.
9. Опишите инжиниринг ПО в инструментальной среде Embarcadero RAD Studio на языке Delphi.
10. Опишите понятие технологии программирования.
11. Какова роль комментирования в технологии программирования?
12. Опишите инжиниринг ПО в инструментальной среде Embarcadero RAD Studio на языке C++.
13. Опишите понятие сопровождения программного средства.
14. Опишите инструментальные системы технологии программирования и их общую архитектуру.
15. Опишите инжиниринг ПО в инструментальной среде MS Visual Studio, мультилингвальная разработка.
16. Опишите понятие качества программного средства.
17. Каковы причины программирования на языке низкого уровня?
18. Опишите инжиниринг ПО в инструментальной среде .Net Framework.
19. Опишите понятие мобильности программного средства.
20. Как осуществляется контроль внешнего описания программного средства?
21. Опишите инжиниринг ПО в инструментальной среде Microsoft Visual C++.
22. Опишите понятие внешнего описания программного средства.
23. В каких случаях применяются компиляторы?
24. Опишите инжиниринг ПО в инструментальной среде Microsoft Visual C#.
25. Опишите понятие устойчивости программного средства.
26. В каких случаях применяются интерпретаторы?
27. Опишите инжиниринг ПО в инструментальной среде Microsoft Visual Basic (classic).
28. Опишите понятие защищенности программного средства.
29. Что такое мобильность программного обеспечения?
30. Опишите инжиниринг ПО в инструментальной среде Microsoft Visual Basic .Net.
31. Опишите понятие коммуникабельности программного средства.
32. Что такое клиент-серверная технология?
33. Опишите инжиниринг ПО в инструментальной среде IDE Symphony.
34. Опишите понятие расширяемости программного средства.
35. Что такое технология облачных вычислений?
36. Опишите инжиниринг ПО в инструментальной среде Wysiwyg-редакторов.
37. Опишите понятие модульности программного средства.
38. Какой должна быть оптимальная длина программного модуля?
39. Опишите инжиниринг ПО в инструментальной среде конструкторов сайтов.
40. Опишите понятие архитектуры программного средства.
41. Что такое правила для установления свойств составного и условного операторов?
42. Опишите инжиниринг ПО в инструментальной среде IDE Net Beans на языке PHP.
43. Что такое правило для установления свойств оператора цикла?
44. Приведите пример среды визуального программирования.
45. Опишите инжиниринг ПО в инструментальной среде IDE Net Beans на языке JavaScript.
46. Опишите понятие программного модуля.
47. Что такое разделяемые библиотеки? Их назначение?
48. Опишите инжиниринг ПО в инструментальной среде MS Silverlight.
49. Опишите понятие отладки программного средства.
50. Что такое программный драйвер?
51. Опишите инжиниринг ПО в инструментальной среде JavaFX.
52. Опишите понятие тестирования программного средства.
53. Перечислите заповеди отладки программных средств.
54. Опишите инжиниринг ПО в инструментальной среде Joomla CMS.

55. Опишите понятие управления конфигурацией программного средства
56. Как производится автономная отладка и тестирование программного средства?
57. Приведите примеры языков высокого уровня.
58. Опишите понятие администратора программного средства.
59. Что такое событийная парадигма в технологии программирования?
60. Опишите инжиниринг ПО в инструментальной среде Drupal CMS.
61. Опишите понятие управления разработкой программного средства.
62. Что такое многозадачность в операционной системе?
63. Опишите инжиниринг ПО в инструментальной среде VRML.
64. Опишите понятие менеджера программного проекта.
65. Для чего применяется технология разработки параллельных вычислений?
66. Опишите инжиниринг ПО в инструментальной среде IDE Geany.
67. Опишите понятие аттестации программного средства.
68. Как достигается обеспечение защиты от несанкционированного доступа к программным средствам и защиты от взлома защиты?
69. Для чего применяется технология функционального программирования?
70. Опишите понятие компьютерной технологии программирования.
71. Как достигается обеспечение легкости применения программного средства?
72. Что такое динамические языки и в чем их преимущество в технологии программирования?
73. Опишите понятие пользовательского объекта в программном средстве.
74. Как достигается обеспечение эффективности программного средства?
75. Опишите инжиниринг ПО в инструментальной среде Wordpress CMS.
76. Опишите понятие инструментальной среды разработки и сопровождения программных средств.
77. Как достигается обеспечение сопровождаемости программного средства?
78. Что такое объектный модуль?
79. Опишите понятие интегрированности инструментальной среды разработки и сопровождения программных средств .
80. Перечислите виды документов программного средства.
81. Что такое библиотечный модуль?
82. Опишите понятие языково-ориентированной инструментальной среды программирования
83. Опишите структуру управления разработкой программного средства.
84. Что такое объектно-ориентированное программирование?
85. Опишите понятие инструментальной системы технологии программирования .
86. Каковы особенности внешнего описания программных средств при объектном подходе к разработке?
87. Приведите примеры операционных систем.
88. Опишите понятие языково-зависимой инструментальной системы технологии программирования
89. Опишите инструментальные среды разработки и сопровождения программных средств. Принципы их классификации.
90. Опишите инжиниринг ПО в инструментальной среде IDE Net Beans на языке C++.
91. Опишите модель перевода и источники ошибок при разработке программных средств.
92. Что такое метод целенаправленной конструктивной реализации?
93. Опишите инжиниринг ПО в инструментальной среде языка Дракон.
94. Каковы специфические особенности разработки программных средств?
95. Что такое структурное программирование и пошаговая детализация?
96. Опишите инжиниринг ПО в инструментальной среде TopSpeed Clarion RAD.
97. Какие требования предъявляются к программному средству?
98. Опишите условия экономической целесообразности программирования.
99. Опишите инжиниринг ПО в инструментальной среде AutoCAD.
100. Что такое жизненный цикл программного средства?
101. Опишите понятие архитектурной функции программного средства.
102. Опишите инжиниринг ПО в инструментальной среде создания и ведения БД Microsoft SQL Server.
103. Что такое спецификация качества программного средства?
104. Опишите понятие о завершаемости выполнения программы.
105. Опишите инжиниринг ПО в инструментальной среде создания и ведения БД СУБД Oracle.
106. Что такое функциональная спецификация программного средства?
107. Как производится комплексная отладка и тестирование программного средства?
108. Опишите инжиниринг ПО в инструментальной среде создания и ведения БД СУБД Линтер.

109. Каковы причины программирования на языке высокого уровня?
110. Что такое линкер, каковы его функции?
111. Опишите инжиниринг ПО в инструментальной среде создания и ведения БД СУБД SAP HANA.
112. Что такое табличный подход к спецификации семантики функций? Опишите метод таблиц решений.
113. Что такое программная эмуляция?
114. Опишите инжиниринг ПО в инструментальной среде создания и ведения БД IBM DB2.
115. Какие основные классы архитектур программных средств применяются в технологии программирования?
116. Что такое препроцессор и какова его роль в технологии программирования?
117. Опишите инжиниринг ПО в инструментальной среде создания ПО 1С:Предприятие.
118. Опишите понятие программного модуля и его основные характеристики.
119. Как достигается обеспечение устойчивости программного модуля?
120. Опишите инжиниринг ПО в инструментальной среде технологии CUDA (в MS Visual Studio)
121. Какие есть методы разработки структуры программ?
122. достигается обеспечение защиты от влияния "чужих" программ?
123. Опишите инжиниринг ПО в инструментальной среде SAP, язык ABAP/4.

### 3.3. Задания для оценки освоения МДК 03.03 Документирование и сертификация

#### 3.3.1 Типовой тест к дифференцированному зачету

1. Метрология - это наука о
  - Субъективном мире человека и животных
  - \* Измерениях
  - Земной коре
  - Взаимодействиях организмов друг с другом и с окружающей средой
  - Фундаментальных основах и закономерностях строения и эволюции мира
2. К области законодательной метрологии не относится
  - Установка требований, направленных на достижение единства измерений
  - разработка и внедрение норм и правил выполнения измерений
  - установка терминов и определений в области метрологии
  - \* изучение проблем измерений в целом и образующих измерение элементов
3. С понятием метрология связана область человеческой деятельности, относящаяся к
  - Экономике
  - Менеджменту
  - Планированию
  - \* Измерениям
4. - это процесс, заключающийся в определении значения физической величины с помощью технических средств
  - \* Измерение
  - Обучение
  - Внедрение
  - Вычисление
5. Конкретная физическая величина, определенная и принятая по соглашению, с которой сравниваются другие величины того же рода
  - \* Единица измерения ФВ
  - Размер ФВ
  - Средство измерения
  - Точность измерений
6. Назовите основные единицы системы СИ
  - \* Кельвин
  - Герц
  - \* Моль
  - Ньютон
  - Джоуль
  - \* Кандела
  - \* Ампер
  - Фарад
  - \* Секунда
7. Назовите причины возникновения погрешностей
  - \* Влияние условий проведения эксперимента
  - \* Человеческий фактор
  - \* Несовершенство метода измерения
  - \* Несовершенство технических средств
8. — это число, отражающее степень точности измерения.
  - Абсолютная погрешность
  - \* Относительная погрешность
  - Методическая погрешность
  - Дополнительная погрешность
  - Статистическая погрешность
9. По характеру проявления погрешности СИ могут быть классифицированы как
  - абсолютные и относительные
  - основные и дополнительные

статические и динамические

\* систематические и случайные

10. По взаимодействию изменений во времени и входной величины погрешности могут быть классифицированы как

абсолютные и относительные

основные и дополнительные

\* статические и динамические

систематические и случайные

11. В основе Международной системы единиц лежат семь единиц, охватывающие следующие области науки:

механика, электричество, теплоту, оптику, молекулярную физику, термодинамику и химию.

Установите соответствие между физической величиной и ее единицей измерения:

Единица длины (механика)

Метр

Единица массы (механика)

Килограмм

Единица времени (механика)

Секунда

Единица силы электрического тока (электричество)

Ампер

Единица термодинамической температуры (теплота)

Кельвин

Единица силы света (оптика)

Кандела

Единица количества вещества (молекулярная физика, термодинамика и химия)

Моль

12. Причиной возникновения погрешности является человеческий фактор

Инструментальной

Статической

\* Субъективной

Динамической

13. Относительная погрешность измеряется в

Метрах

\* Процентах

Пунктах

Килограммах

14. Классификация каких погрешностей изображены на рисунке



\* Систематических

Относительных

Абсолютных

Дискретных

15. При поверке используют...

Линейку

\* Эталон

Рулетку

Калибровку

16. Поверке подвергают средства измерений, подлежащие.

\* Государственному метрологическому контролю и надзору

Калибровке

Воспроизведению основной единицы

Измерению

17. Средства измерений, не подлежащие государственному метрологическому контролю и надзору могут подвергаться.

\* Калибровке

Поверке

Измерению

Изучению

18. Расшифруйте аббревиатуру ГСИ

\* государственная система обеспечения единства измерений

государственная измерительная система

государственная система испытаний

государственная система измерений

19. Эталон должен обладать, по крайней мере, тремя тесно связанными друг с другом признаками:

\* Неизменностью

Субъективностью

\* Воспроизводимостью

\* Сличаемостью

20. Установите соответствие между определениями

Установление органом государственной метрологической службы пригодности СИ к применению на основании экспериментально определяемых метрологических

характеристик и подтверждения их соответствия установленным обязательным требованиям.

Поверка СИ

Средство измерений (или их комплекс), предназначенное для воспроизведения и (или) хранения единицы и передачи ее размера нижестоящим по поверочной схеме СИ и утвержденное в качестве эталона в установленном порядке.

Эталон

Совокупность операций, устанавливающих соотношений между значением величины, полученным с помощью данного средства измерений и соответствующим значением величины, определенным с помощью эталона с целью определения действительных метрологических характеристик этого средства измерений.

Калибровка СИ

21. Выделите виды эталонов

- \* Международный эталон
- \* Первичный эталон
- \* Вторичный эталон
- \* Эталон сравнения
- \* Рабочий эталон
- \* Рабочее средство измерений
- \* Эталонная база страны
- \* Государственный первичный эталон

22. На рисунке изображена

Государственные  
первичные эталоны

Вторичные  
эталон

Рабочие  
эталон

Рабочие средства  
измерений

- \* Структура эталонной базы Российской Федерации

Эталон сравнения

Рабочие средства измерения Международный эталон

23. Совокупность государственных первичных и вторичных эталонов, являющаяся основой обеспечения единства измерений в стране, называется

- \* Эталонная база страны

Вторичный эталон

Первичный эталон

Эталон сравнения

24. Зафиксированная на материальном носителе информация с реквизитами, позволяющими ее идентифицировать:

- 1) Сообщение;
- 2) Документ;
- 3) Текст.

25. Что не относится к свойствам, которыми должно обладать сообщение, чтобы стать документом?
- 1) Легитимность;
  - 2) Доступность;
  - 3) Качество.
26. Что не относится к внутренним признакам документа?
- 1) Стиль;
  - 2) Управляющая сила;
  - 3) Размер.
27. К каким признакам документа относятся: авторство, наименование вида документа, датирование, удостоверение, форма и размер, способ документирования и используемые носители?
- 1) Внешние;
  - 2) Документообразующие;
  - 3) Внутренние.
28. По какому признаку классифицированы следующие документы: протоколы, приказы, отчеты?
- 1) Вид документа;
  - 2) Качество документа;
  - 3) Разновидность документа.
29. По какому признаку классифицирована информация на общедоступную и конфиденциальную?
- 1) Порядок представления;
  - 2) Порядок распространения;
  - 3) Категория доступа.
30. Что не относится к юридически значимым элементам оформления документа?
- 1) Телефон организации, являющейся автором документа;
  - 2) Подпись документа;
  - 3) Регистрационный номер документа.
31. Частное понятие, которое детализирует, уточняет характер деятельности, документируемой тем или иным видом:
- 1) Вид документа;
  - 2) Разновидность документа;
  - 3) Признак документа.
32. Совокупность присущих документу существенных признаков и особенностей, позволяющих выделить его из среды других предметов:
- 1) Качество документа;
  - 2) Признак документа;
  - 3) Вид документа.
33. К чему относится изменение во времени яркости элементов изображения в телевизионных системах?
- 1) Документ;
  - 2) Информация;
  - 3) Сообщение.
34. Информация, содержащая сведения об источниках, где могут находиться необходимые данные (факты):
- 1) Постоянная;
  - 2) Фактографическая;
  - 3) Документальная.
35. Задача документоведения:
- 1) Практическое обоснование процессов документационного обеспечения аппарата управления обществом;



- 2) Теоретическое обоснование процессов документационного обеспечения аппарата управления обществом;
- 3) 1 и 2.

### 3.4. Задания для оценки освоения МДК 03.04 1С: Бухгалтерия

#### 3.4.1 Материал собеседования для дифференцированного зачета

1. Как ввести основные сведения об организации (ответственные лица, контактная информация, банковские счета, система налогообложения)?
2. Как осуществить настройки параметров учета (общие настройки, маркетинг, нормативно-справочная информация, оптовые продажи, розничные продажи, запасы и закупки, финансы, органайзер)?
3. Как добавить новое структурное подразделение в организацию?
4. Как установить новые виды цен?
5. Каким образом осуществляется выбор валюты для ведения экономической деятельности организации?
6. Как зарегистрировать нового партнера (покупателя, поставщика)?
7. Как создать типовое соглашение с клиентом? Для чего оно используется?
8. Как добавить новую позицию номенклатуры в справочник?
9. Каким образом устанавливаются скидки и наценки (в том числе по маркетинговым мероприятиям) на отдельные наименования номенклатуры?
10. Каким образом происходит ввод начальных остатков товаров на складах?
11. Как происходит ввод остатков товаров, принятых и переданных на реализацию?
12. Как осуществляется ввод остатков по взаиморасчетам с покупателями?
13. Как совершить покупку товара с предоплатой поставщику?
14. Как зарегистрировать выписку банка?
15. Как зарегистрировать счет-фактуру при поступлении товара?
16. Как зарегистрировать отгрузку товаров покупателю?
17. Как зарегистрировать заказ товара с резервированием на складах?
18. Как осуществляется возврат товаров?
19. Каким образом происходит инвентаризация товаров на оптовом складе? С какими особенностями можно столкнуться при выполнении данной операции?
20. Как осуществляется процесс регистрации сборки и разборки товаров?
21. Каковы особенности процесса торговли в розницу через автоматизированную торговую точку?
22. Каковы особенности процесса возврата товара в розничной торговле.
23. Каковы особенности торговли в розницу через неавтоматизированную торговую точку.
24. Что такое расходный кассовый ордер? В каких случаях он применяется? Как составляется расходный кассовый ордер?
25. Что такое авансовый отчет? Как он составляется и в каких случаях?
26. Каким образом происходит процесс реализации комиссионного товара?
27. Каковы особенности работы с документами по учету возвратной тары?
28. Для каких целей существует подсистема «Планирование продаж»? Каким образом в ней можно планировать платежи?
29. Для чего компании необходима Книга покупок и Книга продаж?
30. Каким образом составляются отчеты по группам «Продажи», «Запасы и Закупки», «Денежные средства»?

#### 4. Оценка по производственной практике (по профилю специальности)

##### 4.1. Общие положения

Целью оценки по учебной и (или) производственной практике является оценка профессиональных и общих компетенций; практического опыта и умений.

Оценка по учебной и (или) производственной практике выставляется на основании данных аттестационного листа (характеристики профессиональной деятельности обучающегося/студента на практике) с указанием видов работ, выполненных обучающимся во время практики, их объема, качества выполнения в соответствии с технологией и (или) требованиями организации, в которой проходила практика.

##### 4.2. Виды работ практики и проверяемые результаты обучения по профессиональному модулю

###### 4.2.1. Производственная практика (по профилю специальности)

Таблица 4

Виды работ	Проверяемые результаты (ПК, ОК, ПО, У)
Инструктаж о прохождении практики. Знакомство с программой практики и порядок её проведения, изучение правил внутреннего распорядка, знакомство с графиком работы студентов, ведение дневника практики, составление отчета. Ознакомление с порядком сбора информации. Инструктаж по технике безопасности, пожаробезопасности, производственной санитарии под роспись в журнале. Правила безопасности при работе с компьютером.	ОК2 ОК4 ОК5 ОК6
Выбор задания. Анализ постановки задачи. Анализ входных и выходных данных. Определение границ входных и выходных данных. Разработка алгоритма и блок-схемы поставленной задачи. Варианты заданий: Создание диалоговых окон. Создание главных окон. Графика 2D и 3D. Технология «drag-and-drop». Управление компоновкой виджетов на форме. Обработка событий во время продолжительных процессов. Работа с каталогами.	ПК 3.1 ОК 1 ОК 2 ОК 4 ПО1 У1 31 32 37

Разработка кода программного продукта согласно разработанному алгоритму в комплексной среде Qt. Разработка структуры программы. Выделение объектов и определение отношений между	ПК 3.2 ОК 3 У2 34 310
--	-----------------------

объектами. Проектирование классов. Компоновка программных компонентов. Создание виджетов.	311
Отладка кода программного продукта, используя возможности отладчика комплексной среды Qt. Реализация диалога в графическом пользовательском интерфейсе.	ПК 3.3 ОК 7 35
Обследование объекта информатизации. Выбор жизненного цикла программного средства. Определение целей создания ПС. Анализ и разработка требований к ПС. Разработка внешних спецификаций. Системный анализ и проектирование программного средства.	ПК 3.5 ОК 6 39
Документирование в процессе разработки ПС. Стандартизация управления конфигурацией ПС. Разработка руководства пользователя. Измерение и оценка сложности программного средства.	ПК 3.4 ПК 3.6 311
Создание подсистем. Создание справочника. Создание документа. Программирование формы документа. Создание печатной формы документа. Настройка интерфейса пользователя. Использование журналов документов. Использование язык запросов 1С:Предприятие. Использование регистров накопления. Разработка отчетов с использованием системы компоновки данных.	ПК 3.1 ПК 3.3 ОК 4 31 34 36

### 4.3 Форма аттестационного листа

МИНОБРНАУКИ РОССИИ филиал ФГБОУ ВПО «Владивостокский государственный университет экономики и сервиса» г. Артем (колледж)	
<b>АТТЕСТАЦИОННЫЙ ЛИСТ ПО ПРАКТИКЕ</b>	
<b>Абрамов Максим Сергеевич</b>	
<i>ФИО</i>	
обучающийся(аяся) на 3 курсе по специальности СПО 09.02.03 Программирование в компьютерных системах	
успешно прошел(ла) производственную практику(по профилю специальности) по профессиональному модулю:	
<b>ПМ.03 Участие в интеграции программных модулей</b>	
<i>наименование профессионального модуля</i>	
в объеме 76 часов с 30 марта 2015 года по 11 апреля 2015 года	
в организации: филиал ФГБОУ ВПО "ВГУЭС" в г. Артеме	
<b>Виды и качество выполнения работ</b>	
Виды и объем работ, выполненных обучающимся во время практики	Качество выполнения работ в соответствии с технологией и (или) требованиями организации, в которой проходила практика
ПМ.03	
Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения.	
Выполнять интеграцию модулей в программную систему.	
Выполнять отладку программного продукта с использованием специализированных программных средств.	
Осуществлять разработку тестовых наборов и тестовых сценариев.	
Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования.	
Разрабатывать технологическую документацию.	
Итоговая оценка _____	
Дата _____ 201 _ г.	
Руководитель от филиала	/ Будников А.И. /

## 5. Контрольно-оценочные материалы для экзамена (квалификационного)

### 5.1. Паспорт

#### Назначение:

Контрольно-оценочные средства (далее КОМ) предназначены для контроля и оценки результатов освоения профессионального модуля ПМ.03 Участие в интеграции программных модулей по специальности 09.02.03 Программирование в компьютерных системах

### 5.2. Задание для экзаменуемого вариант №

#### Задание 1

**Коды** проверяемых профессиональных и общих компетенций:

#### Инструкция:

Внимательно прочитайте задание.

Вы можете воспользоваться (*указать, чем*)

Время выполнения задания - \_\_\_\_\_

Текст задания: ...

#### Задание n

**Коды** проверяемых профессиональных и общих компетенций:

#### Инструкция:

Внимательно прочитайте задание.

Вы можете воспользоваться (*указать, чем*)

Время выполнения задания - \_\_\_\_\_

Текст задания: .

### 5.3. Пакет экзаменатора

#### 5.3.1. Условия

Количество вариантов каждого задания/пакетов заданий для экзаменуемого:

Время выполнения каждого задания: \_\_\_\_\_

Оборудование: \_\_\_\_\_

Литература для учащегося: \_\_\_\_\_

Учебники: \_\_\_\_\_

Методические пособия: \_\_\_\_\_

Справочная литература: \_\_\_\_\_

#### 5.3.2. Ход выполнения задания

Таблица 5

Коды проверяемых	Показатели оценки результата	Оценка (да/нет)
------------------	------------------------------	-----------------

компетенций		

**2) Подготовленный продукт / осуществленный процесс:**

Таблица 6

Коды проверяемых компетенций	Показатели оценки результата	Оценка (да/нет)



## 6. Оценочная ведомость по профессиональному модулю

### Оценочная ведомость по профессиональному модулю

#### ПМ.03 Участие в интеграции программных модулей

Ф.И.О.,

обучающийся (-аяся) на четвертом курсе по профессии НПО /по специальности СПО 09.02.03

Программирование в компьютерных системах

освоил (-а) программу профессионального модуля ПМ.03 Участие в интеграции программных модулей в объеме **1047** часов с 1 сентября 2014 г. по 13 апреля 2015 г.

#### Результаты промежуточной аттестации по элементам профессионального модуля

Элементы модуля	Формы промежуточной аттестации	Оценка
МДК.03.01 Технология разработки программного обеспечения	7 семестр - другая форма контроля	
	8 семестр - экзамен	
МДК.03.02 Инструментальные средства разработки программного обеспечения	7 семестр - другая форма контроля	
	8 семестр - экзамен	
МДК.03.03 Документирование и сертификация	8 семестр - дифференцированный зачет	
МДК.03.04 1С: Бухгалтерия	8 семестр - дифференцированный зачет	
ПП.03.01 Производственная практика (по профилю специальности)	8 семестр - дифференцированный зачет	
ПМ.03 Участие в интеграции программных модулей	8 семестр - Экзамен (квалификационный)	

Профессиональные и общие компетенции	Показатели оценки результата	Оценка да/нет
ПК 3.1 - Анализировать проектную и техническую документацию на уровне взаимодействия компонент программного обеспечения	Определение характера взаимодействия компонентов программного обеспечения	
	Участие в проектировании программного обеспечения с использованием языка объектно-ориентированного моделирования UML	
	Точность и грамотность оформления технологической документации	
ПК 3.2 - Выполнять интеграцию модулей в программную систему	Владение основными подходами интеграции программных модулей	
	Определение этапов разработки программного обеспечения	
	Демонстрация построения концептуальной, логической и физической моделей баз данных	
	Самостоятельный обоснованный выбор технологии разработки исходного модуля исходя из его назначения	
	Самостоятельный обоснованный выбор методов разработки программных модулей	
	Демонстрация навыков модификации программных модулей	

ПК 3.3 - Выполнять отладку программного продукта с использованием специализированных программных средств	Овладение принципами построения, структурами и приемами работы с инструментальными средствами, поддерживающими отладку программного обеспечения	
	Определение возможности увеличения быстродействия программного продукта	
	Определение способов и принципов оптимизации	
	Выбор методов отладки программных модулей и программного обеспечения	
	Выбор специализированных средств для отладки программного обеспечения	
	Демонстрация устранения ошибок в программных модулях	
ПК 3.4 - Осуществлять разработку тестовых наборов и тестовых сценариев	Тестовый набор для тестирования программного обеспечения на уровне пакета, модуля, класса, компонента	
	Тестовый сценарий для тестирования графического интерфейса	
	Результаты нагрузочного тестирования программного обеспечения	
	Демонстрация навыков правильного использования инструментальных средств тестирования программных модулей	
ПК 3.5 - Производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	Оформление программного кода программного модуля в соответствии с требованиями стилизового оформления	
	Оформление программного кода программного обеспечения в соответствии с требованиями стандартов кодирования	
	Заключение о соответствии программного модуля стандартам качества программного обеспечения	
	Способен производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования	
ПК 3.6 - Разрабатывать технологическую документацию	Выполненная документация для пользователя программного обеспечения	
	Документация к программному коду	
	Контрольные варианты использования программного обеспечения	
	Правильность выбора методов средств разработки программной документации	
	Точность и грамотность оформления технологической документации	
ОК 1 - Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес	Динамика успеваемости по МДК, положительный отзыв руководителя практики.	
	Активное посещение учебных занятий и практики, консультаций.	
ОК 2 - Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество	Мотивированное обоснование выбора и применения методов и способов решения профессиональных задач.	
	Точность, правильность и полнота выполнения профессиональных задач.	
	Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями	
	Разработка пользовательского интерфейса	
ОК 3 - Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	Демонстрация способности принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.	

ответственность	Использование специализированных программных средств отладки программных модулей.	
-----------------	---	--

	Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.	
ОК 4 - Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития	Обоснованность выбора информационных источников для решения профессиональных задач.	
	Оперативность поиска и использования необходимой информации для качественного выполнения профессиональных задач, профессионального и личностного развития.	
	Широта использования различных источников информации, включая электронные.	
	Использование инструментальных средств и графических языков спецификаций для создания компонент проектной и технической документации.	
	Оформление проектной и технической документации в соответствии со стандартами.	
ОК 5 - Использовать информацию - коммуникационные технологии в профессиональной деятельности	Оперативность и широта осуществления операций с использованием общего и специализированного программного обеспечения.	
	Создание отдельных компонент.	
	Выполнение спецификаций компонент.	
ОК 6 - Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями	Результаты выполнения задания на учебной практике.	
	Положительный отзыв руководителя практики.	
	Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями.	
	Разработка пользовательского Интерфейса.	
	Использование специализированных программных средств отладки программных модулей.	
	Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.	
	Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев.	
Определение ошибок в программном коде с использованием тестовых наборов.		
ОК 7 - Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий	Ответственность за результат выполнения заданий на практике.	
	Способность к самоанализу и коррекции результатов собственной работы.	
	Использование специализированных программных средств отладки программных модулей.	
	Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.	
	Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев.	
	Определение ошибок в программном коде с использованием тестовых наборов.	
	Выявление избыточности кода программного продукта и его оптимизация.	
Анализ оптимизации программного кода модуля.		
ОК 8 - Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации	Качество, своевременность и полнота выполнения заданий внеаудиторной самостоятельной работы.	
	Обоснованность постановки целей и задач самообразования.	
	Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями.	

	Разработка пользовательского Интерфейса.	
	Использование специализированных программных средств отладки программных модулей.	
	Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.	
	Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев.	
	Определение ошибок в программном коде с использованием тестовых наборов.	
	Выявление избыточности кода программного продукта и его оптимизация, анализ оптимизации программного кода модуля	
ОК 9 - Ориентироваться в условиях частой смены технологий в профессиональной деятельности	Проявление интереса к инновациям в области профессиональной деятельности	
	Выполнение создания кода программного продукта на уровне модуля в соответствии с готовыми спецификациями	
	Разработка пользовательского интерфейса	
	Использование специализированных программных средств отладки программных модулей.	
	Разработка компонент программных модулей с использованием современных инструментальных средств и технологий.	
	Выполнение тестирования качества разработки программных модулей с помощью разработанных тестовых наборов и сценариев	
	Определение ошибок в программном коде с использованием тестовых наборов.	
	Выявление избыточности кода программного продукта и его оптимизация	
ОК 10 - Исполнять воинскую обязанность, в том числе с применением полученных профессиональных знаний (для юношей)	Анализ оптимизации программного кода модуля.	
	Участие в спортивных мероприятиях колледжа и мероприятиях.	
	Соблюдение правил внутреннего распорядка колледжа.	

Дата «»20 \_\_\_\_\_ г. Подписи членов экзаменационной комиссии