

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РД
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ РД
«КОЛЛЕДЖ ЭКОНОМИКИ И ПРАВА»**

УТВЕРЖДАЮ

Зам. директора по учебной работе

А.Л. Джалилова

А.Л. Джалилова

подпись

ФИО

« 2 » 09 2023 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ**

**ПМ.02 ОСУЩЕСТВЛЕНИЕ ИНТЕГРАЦИИ ПРОГРАММНЫХ
МОДУЛЕЙ**

специальность 09.02.07 «Информационные системы и программирование»

Дербент 2023

Организация – разработчик:

Государственное бюджетное профессиональное образовательное учреждение РД «Колледж экономики и права»

Разработчики:

Гюльмагомедов Т.Х, Мурадханова Д.Б., преподаватель экономических дисциплин ГБПОУ РД «КЭиП»

Одобрена на заседании цикловой (предметной) комиссии профессиональных дисциплин и модулей по специальности 09.02.07 «Информационные системы и программирование»

Протокол № 1 от « 28 » 08 2023 г.

Председатель комиссии  / Гюльмагомедов Т.Х /

Паспорт фонда оценочных средств

Общие положения

Фонд оценочных средств (ФОС) предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу профессионального модуля ПМ.02 Осуществление интеграции программных модулей.

ФОС включает контрольные материалы для проведения текущего контроля и промежуточной аттестации по МДК, учебной и производственной практике и в целом по профессиональному модулю в рамках экзамена по модулю.

Результатом освоения профессионального модуля является готовность обучающегося к выполнению вида профессиональной деятельности **осуществление интеграции программных модулей** и составляющих его профессиональных компетенций, а также общие компетенции, формирующиеся в процессе освоения ОПОП в целом.

Формой аттестации по профессиональному модулю является экзамен по модулю.

Фонд оценочных средств разработан на основании:

1. ФГОС СПО 09.02.07 Информационные системы и программирование
2. Примерной основной образовательной программы по специальности 09.02.07 Информационные системы и программирование
3. Рабочей программы профессионального модуля ПМ.02 Осуществление интеграции программных модулей.
4. Учебного плана по специальности 09.02.07 Информационные системы и программирование
5. Положения о формах, периодичности и порядке текущего контроля успеваемости и промежуточной аттестации обучающихся
6. Положения о квалификационном экзамене
7. Шаблона фонда оценочных средств профессионального модуля

1. Результаты освоения профессионального модуля, подлежащие проверке

1.1. Профессиональные и общие компетенции

В результате контроля и оценки по профессиональному модулю осуществляется комплексная проверка следующих профессиональных и общих компетенций:

1.2. Показатели сформированности профессиональных компетенций

Таблица 1

Профессиональные компетенции	Показатели оценки результата	№ заданий для проверки
ПК 2.1 Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент	<ul style="list-style-type: none"> - разработан и обоснован вариант интеграционного решения с помощью графических средств среды разработки, - указано хотя бы одно альтернативное решение; - бизнес-процессы учтены в полном объеме; - вариант оформлен в полном соответствии с требованиями стандартов; - результаты верно сохранены в системе контроля версий. 	1-22
ПК 2.2 Выполнять интеграцию модулей в программное обеспечение	<ul style="list-style-type: none"> - проанализирована архитектура проекта, архитектура доработана для интеграции нового модуля; - выбраны способы форматирования данных и организована их постобработка, транспортные протоколы и форматы сообщений обновлены (при необходимости); - протестирована интеграция модулей проекта и выполнена отладка проекта с применением инструментальных средств среды; - выполнена доработка модуля и дополнительная обработка исключительных ситуаций в том числе с созданием классов-исключений (при необходимости); - определены качественные показатели полученного проекта; - результат интеграции сохранен в системе контроля версий. 	5,7, 11-15,21
ПК 2.3 Выполнять отладку программного модуля с использованием специализированных программных средств ПК	<ul style="list-style-type: none"> - протестирована интеграция модулей проекта и выполнена отладка проекта с применением инструментальных средств среды; - проанализирована и сохранена отладочная информация; 	3,4,6,7, 11-15, 21

	<ul style="list-style-type: none"> - выполнена условная компиляция проекта в среде разработки; - определены качественные показатели полученного проекта в полном объеме; - результаты отладки сохранены в системе контроля версий. 	
ПК 2.4 Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения	<ul style="list-style-type: none"> - обоснован размер тестового покрытия; - разработан тестовый сценарий и тестовые пакеты в соответствии с этим сценарием в соответствии с минимальным размером тестового покрытия; - выполнено тестирование интеграции и ручное тестирование; - выполнено тестирование с применением инструментальных средств; - выявлены ошибки системных компонент (при наличии); - заполнены протоколы тестирования. 	2-5,7,10, 11-19, 21
ПК 2.5 Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования	<ul style="list-style-type: none"> - продемонстрировано знание стандартов кодирования более чем одного языка программирования, - выявлены все имеющиеся несоответствия стандартам в предложенном коде. 	5,7,15

1.3. Показатели сформированности общих компетенций

Таблица 2

Общие компетенции	Показатели оценки результата	№ заданий для проверки
ОК 1. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	<ul style="list-style-type: none"> - Признание наличия проблемы. - Выстраивание вариантов альтернативных действий в случае возникновения нестандартных ситуаций. - Выбор оптимальных способов решения профессиональных задач применительно к различным контекстам. - Определение возможных рисков и методов их снижения при выполнении профессиональных задач. 	1-22

<p>ОК 2. Использовать современные средства поиска, анализа и интерпретации информации, и информационные технологии для выполнения задач профессиональной деятельности.</p>	<ul style="list-style-type: none"> - Нахождение и использование разнообразных источников информации, включая интернетресурсы. - Грамотное определение типа и формы необходимой информации. - Получение нужной информации и сохранение ее в удобном для работы формате. - Извлечение ключевых фрагментов и основного содержания из всего массива информации. - Соответствие выбранной информации действующему законодательству 	<p>1-22</p>
<p>ОК 3. Планировать и реализовывать собственное профессиональное и личностное развитие,</p>	<ul style="list-style-type: none"> - Планирование и способность к организации самостоятельных занятий и домашней работы при изучении профессионального модуля. 	<p>1-22</p>
<p>предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях.</p>	<ul style="list-style-type: none"> - Своевременность сдачи практических заданий, отчетов по практике; - Рациональность распределения времени при выполнении практических работ с соблюдением норм и правил внутреннего распорядка. - Эффективный поиск возможностей развития профессиональных навыков. - Планирование направлений повышения квалификации, совершенствование плана личностного развития. 	
<p>ОК 4. Эффективно взаимодействовать и работать в коллективе и команде.</p>	<ul style="list-style-type: none"> - Взаимодействие с коллегами, руководством, клиентами, самоанализ и коррекция результатов собственной работы. - Правильная оценка вклада членов команды в общекорпоративную работу. - Передача информации, идей и опыта членам команды. - Формирование понимания членами команды личной и коллективной ответственности. - Демонстрация навыков эффективного общения. 	<p>2-7</p>
<p>ОК 5. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста</p>	<ul style="list-style-type: none"> - Использование механизмов создания и обработки текста. - Демонстрация навыков ведения деловых бесед и участия в совещаниях. - Демонстрация навыков владения деловой телефонной коммуникацией. 	<p>1-22</p>

<p>ОК 6. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей</p>	<ul style="list-style-type: none"> - Участие в конференциях, конкурсах, дискуссиях и других образовательных и профессиональных мероприятиях. - Демонстрация своих профессиональных качеств в деловой и доброжелательной форме. - Проявление активной жизненной позиции. - Демонстрация навыков общения в коллективе в соответствии с общепринятыми нормами поведения. 	<p>1-22</p>
<p>ОК 7. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях.</p>	<ul style="list-style-type: none"> - Содействие ресурсосбережению. - Демонстрация навыков эффективных действий в чрезвычайных ситуациях. - Соблюдение норм экологической безопасности и определения направлений ресурсосбережения в рамках профессиональной деятельности. 	<p>1-22</p>
<p>ОК 8. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности</p>	<ul style="list-style-type: none"> - Развитие спортивного воспитания. - Демонстрация успешного выполнения нормативов Всероссийского физкультурно-спортивного комплекса "Готов к труду и обороне" (ГТО). - Содействие укреплению здоровья и профилактике общих и профессиональных заболеваний, пропаганде здорового образа жизни. 	<p>1-22</p>
<p>ОК 9. Пользоваться профессиональной документацией на государственном и иностранном языках.</p>	<ul style="list-style-type: none"> - Грамотное применение специализированного программного обеспечения для сбора, хранения и обработки бухгалтерской информации. - Правильная интерпретация интерфейса специализированного программного обеспечения и нахождение контекстной помощи. - Эффективное применение методов и средств защиты бухгалтерской информации. - Выбор информационной системы в условиях конкретной ситуации; - Создание, редактирование и хранение документов в ходе выполнения практических заданий с применением информационных технологий 	<p>1-22</p>

1.4. Дидактические единицы «иметь практический опыт», «уметь» и «знать»

В результате освоения программы профессионального модуля обучающийся должен освоить следующие дидактические единицы.

Таблица 3

Коды	Наименования	Показатели оценки результата	№ заданий для проверки
Иметь практический опыт:			
ПО 1	Модели процесса разработки программного обеспечения;	- анализ архитектуры проекта, архитектура доработана для интеграции нового модуля - демонстрация построения концептуальной, логической и физической моделей программного обеспечения и отдельных модулей; - точность и грамотность оформления технологической документации.	1-6,8-10, 15-21
ПО 2	основные принципы процесса разработки программного обеспечения;	- демонстрация навыков модификации программных модулей. - демонстрация устранения ошибок в программных модулях; - выявление ошибок в программных модулях;	2-5, 11-14
ПО 3	основные подходы к интегрированию программных модулей;	- демонстрация навыков внесения изменения в программные модули для обеспечения качества программного обеспечения;	5,7, 11-14
ПО 4	основы верификации и аттестации программного обеспечения	- разработка тестовых наборов и тестовых сценариев; - демонстрация навыков правильного использования инструментальных средств тестирования программных модулей. - демонстрация использования методов тестирования программного обеспечения; - демонстрация навыков использования программных средств для отладки программного продукта.	3,4,6,7, 11-19
Уметь:			
У 1	использовать выбранную систему	- определение качественных показателей полученного проекта в полном объеме;	2-5,7,10-19
	контроля версий;	- выбор верной версии проекта в системе контроля версий; - выбор средств разработки программных модулей;	

У 2	использовать методы для получения кода с заданной функциональностью и степенью качества	- выбор методов обеспечения качества и надежности в процессе разработки сложных программных средств; - способность производить инспектирование компонент программного продукта на предмет соответствия стандартам кодирования. - выбор специализированных средств и методов отладки программных модулей и программного продукта;	2-22
Знать:			
З 1	модели процесса разработки программного обеспечения;	- анализ требований к программному обеспечению; - анализ проектной и технической документации на уровне взаимодействия компонент программного обеспечения;	1-22
З 2	основные принципы процесса разработки программного обеспечения;	- определение этапов разработки программного обеспечения; - выбор средств и методов разработки программных модулей; - определение возможности увеличения быстродействия программного продукта;	1-22
З 3	основные подходы к интегрированию программных модулей;	- определение характера взаимодействия компонентов программного обеспечения; - выбор технологии разработки исходного модуля исходя из его назначения;	1-22
З 4	основы верификации и аттестации программного обеспечения	- изложение основных принципов тестирования; - определение способов и принципов оптимизации;	1-22

Формы промежуточной аттестации по профессиональному модулю

Таблица 4

Элемент модуля	Формы промежуточной аттестации
МДК 02.01,	экзамен
МДК 02.02	экзамен
МДК 02.03	Дифференцированный зачет
УП	Дифференцированный зачет
ПП	Дифференцированный зачет
ПМ (в целом)	Экзамен по модулю

Перечень оценочных средств

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
1.	Устный опрос	Средство, позволяющее оценить знания и кругозор обучающегося, умение логически построить ответ, владение монологической речью и иные коммуникативные навыки; позволяет выявить детали, которые оказались недостаточно осмысленными в ходе учебных занятий и при подготовке к экзамену.	Перечень вопросов, выносимых на опрос
2.	Лабораторно практических работы	метод обучения, при котором обучающиеся под руководством преподавателя и по заранее намеченному плану продельывают опыты или выполняют определенные практические задания.	Практические задания
3.	Курсовая работа	самостоятельная учебная научно-методическая работа студента, выполняемая под руководством преподавателя. Имеет целью развитие у студентов навыков самостоятельной творческой работы, овладение методами современных научных исследований, углублённое изучение какого-либо вопроса, темы, раздела учебной дисциплины	Темы курсовых работ
4.	Тест	Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося.	Фонд тестовых заданий
5.	Экзамен квалификационный	Средство проверки знаний и умений, полученных в результате изучения дисциплины	задание на квалификационный экзамен

Контрольные вопросы для дифференцированного зачета

по профессиональному модулю

«Осуществление интеграции программных модулей»

1. Понятия требований, классификация, уровни требований. Методологии и стандарты, регламентирующие работу с требованиями.
2. Современные принципы и методы разработки программных приложений.
3. Методы организации работы в команде разработчиков. Системы контроля версий
4. Основные подходы к интегрированию программных модулей.
5. Стандарты кодирования.
6. Описание требований: унифицированный язык моделирования. Диаграммы UML.
7. Описание и оформление требований (спецификация). Анализ требований и стратегии выбора решения

8. Цели и задачи и виды тестирования. Стандарты качества программной документации. Меры и метрики.
9. Тестовое покрытие.
10. Тестовый сценарий, тестовый пакет.
11. Анализ спецификаций. Верификация и аттестация программного обеспечения.
12. Понятие репозитория проекта, структура проекта.
13. Виды, цели и уровни интеграции программных модулей. Автоматизация бизнес-процессов.
14. Выбор источников и приемников данных, сопоставление объектов данных.
15. Транспортные протоколы. Стандарты форматирования сообщений.
16. Организация работы команды в системе контроля версий.
17. Отладка программных продуктов. Инструменты отладки. Отладочные классы.
18. Ручное и автоматизированное тестирование.
19. Методы и средства организации тестирования.
20. Инструментарии анализа качества программных продуктов в среде разработки.
21. Обработка исключительных ситуаций.
22. Методы и способы идентификации сбоев и ошибок.
23. Выявление ошибок системных компонентов.
24. Понятие решения. Множество решений, оптимальное решение.
25. Показатель эффективности решения
26. Математические модели, принципы их построения, виды моделей.
27. Задачи: классификация, методы решения, граничные условия.
28. Общий вид и основная задача линейного программирования.
29. Симплекс – метод.
30. Транспортная задача. Методы нахождения начального решения транспортной задачи.
31. Общий вид задач нелинейного программирования.
32. Графический метод решения задач нелинейного программирования.
33. Метод множителей Лагранжа.
34. Основные понятия динамического программирования: шаговое управление, управление операцией в целом, оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию, аддитивный критерий, мультипликативный критерий.
35. Простейшие задачи, решаемые методом динамического программирования.
36. Системы массового обслуживания: понятия, примеры, модели.
37. Метод имитационного моделирования. Единичный жребий и формы его организации. Примеры задач
38. Понятие прогноза. Количественные методы прогнозирования: скользящие средние, экспоненциальное сглаживание, проектирование тренда.

Критерии оценки:

- оценку «отлично» заслуживает обучающийся, показавший всестороннее систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой; как правило, оценка «отлично» выставляется обучающимся, усвоившим взаимосвязь основных понятий профессионального модуля и их

значение для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

- оценку **«хорошо»** заслуживает обучающийся, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе; как правило, оценка «хорошо» выставляется обучающимся, показавшим систематический характер знаний по профессиональному модулю и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;

- оценку **«удовлетворительно»** заслуживает обучающийся, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой; как правило, оценка «удовлетворительно» выставляется обучающимся, допустившим погрешности в ответе на зачете, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

- оценка **«неудовлетворительно»** выставляется обучающемуся, обнаружившему проблемы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий; как правило, оценка «неудовлетворительно» ставится обучающимся, которые не могут продолжать обучение или приступить к профессиональной деятельности по окончании учебного заведения без дополнительных занятий по соответствующему междисциплинарному курсу.

Фонд тестовых заданий

по профессиональному модулю

«Осуществление интеграции программных модулей»

К основным приемам быстрой разработки прототипа ИС относятся ...

- А) разработка приложения итерациями
- Б) использование большого количества технических средств
- В) использование автоматических генераторов (мастеров)
- Г) высокая параллельность работ
- Д) повторное использование частей проекта
- Е) низкая производительность проектировщиков

Критерием выбора алгоритмического языка при проектировании программного обеспечения задачи является ...

- А) объем памяти, занимаемой разработанной программой
- Б) синтаксическая и семантическая ясность языка
- В) совместимость с другими языками
- Г) возможность управления файлами
- Д) решение заказчика проекта
- Е) время написания программы

Модель - это

- а) средство достижения цели;

б) объект-заменитель, который в определенных условиях может заменять объект оригинал, воспроизводя интересующие нас свойства и характеристики оригинала; в) субъективный образ несуществующего, но желаемого состояния среды;

г) совокупность взаимосвязанных элементов, обособленная от среды и взаимодействующая с ней как целое. **Основная идея стандарта IDEF0** – а) построение модели состава;

б) построение модели структуры;

в) построение древовидной функциональной модели;

г) нет правильного ответа

Вход - это

а) данные или объекты, потребляемые или изменяемые функцией;

б) основной результат деятельности функции, конечный продукт

в) стратегии и процедуры, которыми руководствуется функция;

г) необходимые ресурсы

Механизм - это

а) данные или объекты, потребляемые или изменяемые функцией

б) основной результат деятельности функции, конечный продукт;

в) стратегии и процедуры, которыми руководствуется функция;

г) необходимые ресурсы

Точка зрения формулируется при создании

а) контекста;

б) первого уровня декомпозиции;

в) при создании последнего уровня декомпозиции;

г) нет правильного ответа

Функции в моделях IDEF0 и IDEF3 именуются:

а) существительными;

б) глаголами и глагольными фразами;

в) прилагательными;

г) нет правильного ответа

Диаграммы IDEF3 используются для

а) описания логики взаимодействия информационных потоков;

б) описания документооборота и обработки информации;

в) описания структуры системы;

г) нет правильного ответа

Накопители присутствуют в моделях

а) IDEF0

б) DFD

в) IDEF3

г) нет правильного ответа

Внутренние стрелки в моделях IDEF служат для

а) описания взаимодействия системы с внешним миром;

б) описания взаимодействия функций внутри системы между собой;

в) описания взаимодействия функций с накопителями;

г) нет правильного ответа

VRwin предоставляет для оценки модели следующие инструменты

а) стоимостной анализ (ABC)

- б) свойства, определяемые пользователем (UDP)
- в) социальная оценка;
- г) нет правильного ответа

Центры затрат - это

- а) причина, по которой работа выполняется (основной выход работы)
- б) характеристики входов и управлений работы, которые влияют на то, как выполняется и как долго длится работа
- в) статьи расхода по данному предприятию
- г) нет правильного ответа

При вычислении затрат родительской работы

- а) вычисляется произведение затрат времени дочерней работы на частоту работы, затем результаты складываются
- б) вычисляется сумма затрат времени дочерних работ, затем результат умножается на частоту
- в) вычисляется сумма затрат времени дочерних работ
- г) нет правильного ответа

Внешние

сущности

- а) описывают движение объектов из одной части системы в другую
- б) изображают входы и (или) выходы из системы
- в) изображают объекты в покое
- г) нет правильного ответа

В диаграммах

- а) возможны прямые взаимодействия между накопителями
- б) не возможны прямые взаимодействия между накопителями
- в) возможны прямые взаимодействия между накопителями и внешними сущностями
- г) не возможны прямые взаимодействия между накопителями и внешними сущностями

Внешние сущности в моделях именуется

а) существительными

- б) глаголами и глагольными формами
- в) прилагательными
- г) нет правильного ответа

Внутреннюю целостность системы характеризует модель

- а) состава
- б) «черного ящика»
- в) структуры
- г) нет правильного ответа

Развитие организации связано

- а) только с увеличением материальных ресурсов
- б) только с увеличением информационных ресурсов
- в) не столько с наличными ресурсами, сколько с умением их использовать
- г) нет правильного ответа

Описание функциональности в целом, без подробностей, называется

- а) управлением
- б) механизмом
- в) контекстом
- г) нет правильного ответа

Выход – это

- а) данные или объекты, потребляемые или изменяемые функцией
- б) основной результат деятельности функции, конечный продукт
- в) стратегии и процедуры, которыми руководствуется функция
- г) необходимые ресурсы

Цель моделирования формулируется при создании

- а) контекста
- б) первого уровня декомпозиции
- в) при создании последнего уровня декомпозиции
- г) нет правильного ответа

Область описания системы – это

- а) описание того, с кем взаимодействует система
- б) описание того, что будет рассматриваться как компонент системы, а что как внешнее воздействие
- в) описание структуры системы
- г) нет правильного ответа

Стрелки в моделях IDEF0 и DFD именуется

- а) существительными
- б) глаголами и глагольными формами
- в) прилагательными
- г) нет правильного ответа

Туннелирование стрелок в моделях IDEF0 применяется при

- а) изображении малозначительных стрелок
- б) использовании какой-либо стрелки верхнего уровня во всех без исключения работах нижнего уровня
- в) использовании какой-либо стрелки нижнего уровня во всех без исключения работах верхнего уровня
- г) нет правильного ответа

Объект затрат- это

- а) причина, по которой работа выполняется (основной выход работы)
- б) характеристики входов и управлений работы, которые влияют на то как выполняется и как долго длится работа
- в) статьи расхода по данному предприятию
- г) нет правильного ответа

При вычислении затрат времени родительской работы

- а) вычисляется произведение затрат времени дочерней работы на частоту работы, затем результаты складываются
- б) вычисляется сумма затрат времени дочерних работ, затем результат умножается на частоту

- в) вычисляется сумма затрат времени дочерних работ
- г) нет правильного ответа

На контексте DFD изображаются а)

- а) функция обработки информации
- б) перекрестки
- в) внешние сущности
- г) объекты ссылки

Внешнюю целостность системы характеризует модель

- а) состава
- б) «черного ящика»
- в) структуры
- г) нет правильного ответа

Взаимодействие с окружающим миром в нотации IDEF0 описывается в терминах

- а) управления
- б) входа
- в) выхода
- г) механизма

Управление – это

- а) данные или объекты, потребляемые или изменяемые функцией
- б) основной результат деятельности функции, конечный продукт
- в) стратегии и процедуры, которыми руководствуется функция
- г) необходимые ресурсы

Область описания системы формулируется при создании

- а) контекста
- б) первого уровня декомпозиции
- в) при создании последнего уровня декомпозиции
- г) нет правильного ответа

Функции в моделях IDEF0 и DFD называются

- а) потоками
- б) работами
- в) внешними сущностями
- г) накопителями

Диаграммы DFD используются для

- а) описания логики взаимодействия информационных потоков
- б) описания документооборота и обработки информации
- в) описания структуры системы
- г) нет правильного ответа

Внешние сущности присутствуют в моделях

- а) IDEF0
- б) DFD
- в) IDEF3

г) нет правильного ответа

Граничные стрелки в моделях IDEF0 служат для

- а) описания взаимодействия системы с внешним миром
- б) описания взаимодействия функций внутри системы между собой
- в) описания взаимодействия функций с накопителями
- г) нет правильного ответа

Движитель затрат – это

- а) причина, по которой работа выполняется (основной выход работы)
- б) характеристики входов и управлений работы, которые влияют на то, как выполняется и как долго длится работа
- в) статьи расхода по данному предприятию
- г) нет правильного ответа

Общие затраты по работе рассчитываются как

- а) произведение по всем центрам затрат
- б) среднее арифметическое по всем центрам затрат
- в) сумма по всем центрам затрат
- г) нет правильного ответа

Накопители –

- а) описывают движение объектов из одной части системы в другую
- б) изображают входы и (или) выходы из системы
- в) изображают объекты в покое
- г) нет правильного ответа

Общие затраты времени по работе рассчитываются как

- а) произведение затрат времени на единицу работы и частоты выполнения работы
- б) частное затрат времени на работу и частоты выполнения работы
- в) сумма затрат времени на единицу работы и частоту выполнения работы
- г) нет правильного ответа

В качестве критерия оценки моделей, встроенных в VRwin, используются:

- а) стоимость функций
- б) время выполнения функций
- в) производительность работы
- г) автоматизированность функций

В качестве нижнего уровня декомпозиций в модели IDEF0 можно использовать диаграммы нотаций: а) только IDEF0

- б) только DFD
- в) только IDEF3
- г) диаграммы любых нотаций

Буква «U» в аббревиатуре «UML» означает:

- A) United
- Б) Unified
- B) Universal

Модель UML состоит из (укажите лишнее):

- A) сущностей
- Б) отношений

В) множеств

Сущности UML подразделяются на (укажите лишнее)

- А) структурные
- Б) поведенческие
- В) графические
- Г) группирующие
- Д) аннотационные

Отношения UML подразделяются на (укажите лишнее)

- А) зависимости
- Б) ассоциации
- В) уточнения
- Г) обобщения
- Д) реализации

Структурные сущности UML включают в себя (укажите лишнее)

- А) классы
- Б) узлы
- В) пакеты
- Г) варианты использования
- Д) интерфейсы

Поведенческие сущности UML включают в себя (укажите лишнее)

- А) состояния
- Б) деятельности
- В) варианты использования
- Г) интерфейсы

Сущностями UML являются (укажите лишнее)

- А) классы
- Б) узлы
- В) зависимости
- Г) примечания
- Д) варианты использования

Группирующие сущности UML включают в себя

- А) классы Б) узлы
- В) пакеты
- Г) примечания

Аннотационные сущности UML включают в себя

- А) классы Б) узлы
- В) пакеты
- Г) примечания

Отношения зависимости в UML являются

- А) симметричными
- Б) антисимметричными
- В) транзитивными

Отношения обобщения в UML являются

- А) симметричными

- Б) антисимметричными
- В) транзитивными

Отношения ассоциации (без дополнений) в UML являются

- А) симметричными
- Б) антисимметричными
- В) транзитивными

Отношения реализации в UML являются

- А) симметричными
- Б) антисимметричными
- В) транзитивными

Множество канонических диаграмм UML

- А) определяется стандартом языка
- Б) является соглашением пользователей языка В)
- определяется производителями инструментов,
поддерживающих UML

**Множество канонических структурных диаграмм UML включает в себя
(укажите лишнее)**

- А) Диаграммы классов
- Б) Диаграммы использования
- В) Диаграммы компонентов
- Г) Диаграммы объектов

Множество канонических структурных диаграмм UML включает в себя

- А) Диаграммы последовательности
- Б) Диаграммы (кооперации) коммуникации
- В) Диаграммы использования
- Г) Диаграммы размещения

**Множество канонических поведенческих диаграмм UML включает в себя
(укажите лишнее)**

- А) Диаграммы состояний
- Б) Диаграммы деятельности
- В) Диаграммы последовательности
- Г) Диаграммы потоков данных

Множество канонических поведенческих диаграмм UML включает в себя

- А) Диаграммы классов
- Б) Диаграммы компонентов
- В) Диаграммы последовательности
- Г) Диаграммы размещения (развертывания)

Множество канонических диаграмм UML включает в себя (укажите лишнее)

- А) Диаграммы классов
- Б) Диаграммы состояний
- В) Диаграммы последовательности
- Г) Диаграммы потоков данных

Канонические диаграммы использования предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы реализации предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы классов предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы взаимодействия предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы объектов предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы состояний предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы последовательности предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы кооперации предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы размещения предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы деятельности предназначены для описания

- А) поведения
- Б) использования
- В) структуры

Канонические диаграммы компонентов предназначены для описания

- А) поведения
- Б) использования
- В) структуры

На диаграмме использования UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

На диаграмме использования UML применяют следующие основные типы отношений между действующими лицами

- А) Зависимость
- Б) Обобщение
- В) Ассоциация
- Г) Реализация

На диаграмме использования UML применяют следующие основные типы отношений между вариантами использования

- А) Зависимость
- Б) Обобщение
- В) Ассоциация
- Г) Реализация

На диаграмме использования UML применяют следующие основные типы отношений между действующими лицами и вариантами использования

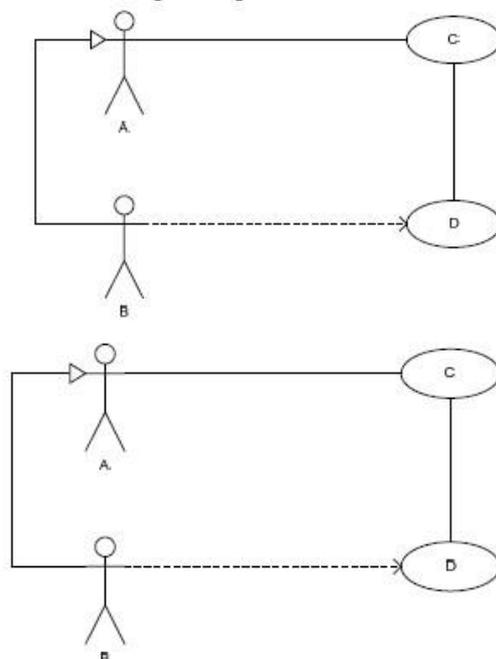
- А) Зависимость
- Б) Обобщение
- В) Ассоциация
- Г) Реализация

Графический комментарий «границы системы» применяется на

- А) Диаграммах классов
- Б) Диаграммах использования
- В) Диаграммах состояний
- Г) Диаграммах деятельности

На рисунке следующие отношения являются недопустимыми на канонической диаграмме

- А) Ассоциация А–С
- Б) Обобщение А–В
- В) Ассоциация С–D
- Г) Все допустимы



На рисунке следующие отношения являются допустимыми на канонической диаграмме Исползования

- А) Ассоциация А–С
- Б) Зависимость В–D
- В) Ассоциация С–D
- Г) Все недопустимы

На диаграмме классов UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

На диаграмме классов UML применяют следующие основные типы отношений между классами

- А) Зависимость
- Б) Обобщение
- В) Ассоциация
- Г) Реализация

На диаграмме классов UML применяют следующие основные типы отношений между интерфейсами

- А) Зависимость
- Б) Обобщение
- В) Ассоциация
- Г) Реализация

На диаграмме классов UML применяют следующие основные типы отношений между интерфейсами и классами

- А) Зависимость
- Б) Обобщение
- В) Ассоциация
- Г) Реализация

На диаграмме компонентов UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

На диаграмме размещения (развертывания) UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

Чтобы показать, что класс является абстрактным, в UML применяется

- А) Подчеркивание имени класса
- Б) Курсивное начертание имени класса
- В) Полужирное начертание имени класса
- Г) Стереотип «abstract»

Имя стереотипа в UML выделяется

- А) Подчеркиванием
- Б) Курсивом
- В) Полужирным начертанием
- Г) Кавычками « »

Дополнительные элементы нотации (украшения) пользователь UML может

- А) включать или не включать в модель
- Б) показывать или не показывать на диаграмме
- В) устанавливать или не устанавливать в инструменте

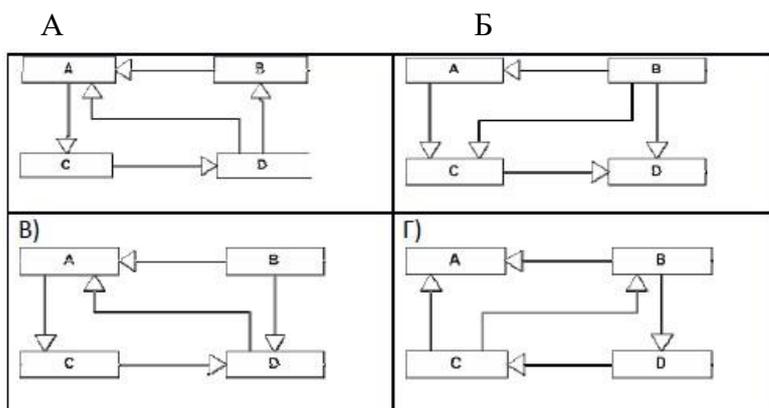
Классификаторами в UML являются (укажите лишнее)

- А) класс
- Б) интерфейс
- В) тип данных
- Г) узел
- Д) компонент
- Е) действующее лицо
- Ж) вариант использования
- З) состояние

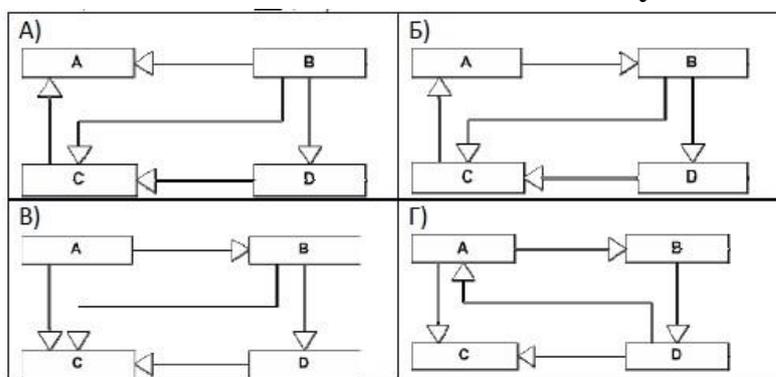
Если классификатор А является обобщением классификатора В, то

- А) Всякий экземпляр классификатора А является экземпляром классификатора В
- Б) Всякий экземпляр классификатора В является экземпляром классификатора А
- В) Всякий прямой экземпляр классификатора В является косвенным экземпляром классификатора А
- Г) Всякий косвенный экземпляр классификатора А является прямым экземпляром классификатора В

Если A, B, C, D — классификаторы, то следующие системы обобщений являются допустимыми



Если A, B, C, D — классификаторы, то следующие системы обобщений являются недопустимыми



Если имя атрибута классификатора подчеркнуто, то

- А) этот атрибут не меняет своего значения
- Б) этот атрибут является атрибутом объекта
- В) все экземпляры данного классификатора имеют одно значение этого атрибута
- Г) этот атрибут является ключевым

Кратность в UML является свойством (укажите лишнее)

- А) классификатора
- Б) полюса ассоциации
- В) операции
- Г) атрибута

Видимость в UML не является свойством

- А) классификатора
- Б) полюса ассоциации
- В) операции
- Г) атрибута
- Д) примечания

Имени в UML не имеют

- А) классы

- Б) пакеты
- В) переходы
- Г) состояния **Стандартными разделами класса в UML не являются**

- А) раздел имени
- Б) раздел свойств
- В) раздел атрибутов
- Г) раздел операций

Обязательными разделами класса в UML являются

- А) раздел имени
- Б) раздел свойств
- В) раздел атрибутов
- Г) раздел операций

Стереотип не может быть указан для

- А) Перехода
- Б) Класса
- В) Действующего лица
- Г) Операции

Если для операции класса указано свойство {isQuery}, то это значит, что

- А) операция реализует запрос к базе данных
- Б) операция не имеет побочных эффектов
- В) операция не может выполняться параллельно с другими операциями **Имя ассоциации**
- А) образуется из имен ассоциированных классов
- Б) образуется из имен ролей ассоциированных классов
- В) указывается в виде текста над линией ассоциации

Если кратность полюса ассоциации задана символом * , то это означает, что

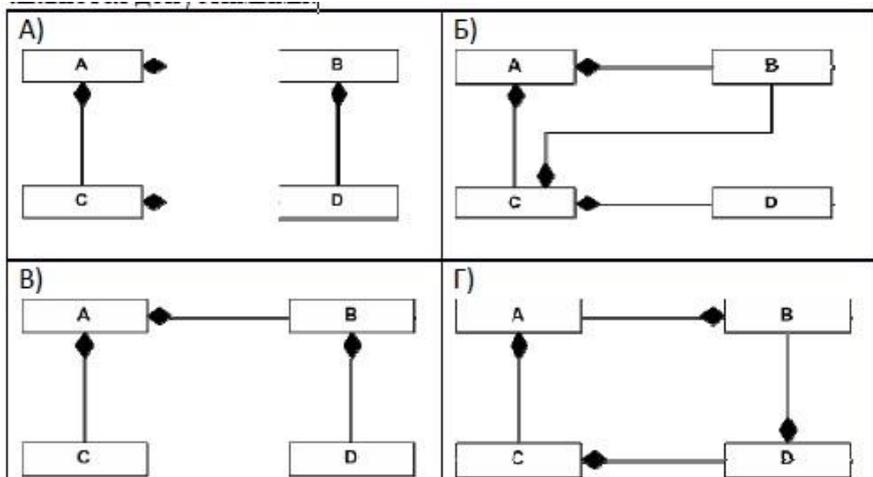
- А) ни одного экземпляра классификатора на данном полюсе ассоциации не участвует в связях, порождаемых ассоциацией
- Б) по меньшей мере один экземпляр классификатора на данном полюсе ассоциации участвует в связях, порождаемых ассоциацией
- В) неопределенное количество экземпляров классификатора на данном полюсе ассоциации участвует в связях, порождаемых ассоциацией

Если кратность полюса ассоциации задана символами 0..1, то это означает, что

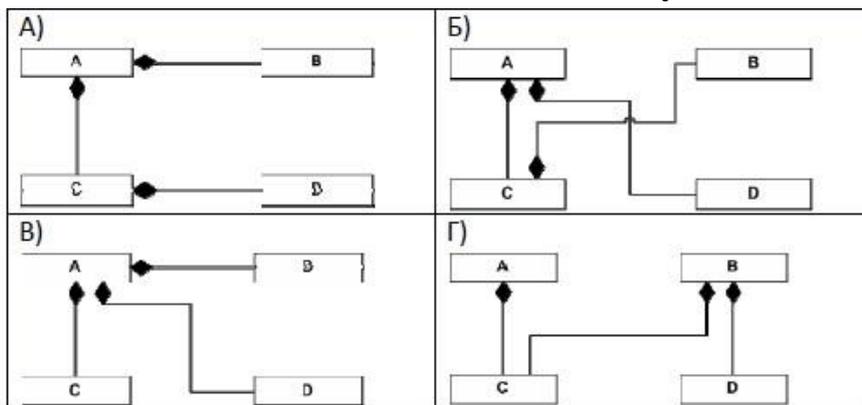
- А) ни одного экземпляра классификатора на данном полюсе ассоциации не участвует в связях, порождаемых ассоциацией
- Б) по меньшей мере один экземпляр классификатора на данном полюсе ассоциации участвует в связях, порождаемых ассоциацией
- В) не более одного экземпляра классификатора на данном полюсе ассоциации участвует в связях, порождаемых ассоциацией

Если в ассоциации классификаторов А и В на полюсе А указан значок композиции (закрашенный ромбик), то это означает что

- А) экземпляры класса В порождаются экземплярами класса А
 - Б) время жизни экземпляров класса В совпадает с временем жизни экземпляров класса А
 - В) всякий прямой экземпляр класса В является косвенным экземпляром класса А
- Если А, В, С, D — классы, то следующие системы композиций являются допустимыми:**



Если А, В, С, D — классы, то следующие системы композиций являются недопустимыми

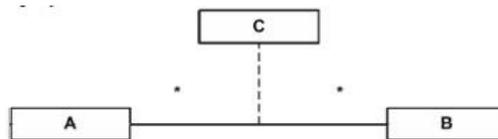


Если у полюса ассоциации указан квалификатор с кратностью

- 0..1, то это означает что
- А) кратность полюса равна 1
- Б) кратность полюса равна 0
- В) кратность полюса равна *
- Г) кратность полюса неизвестна

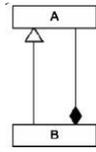
На рисунке С является

- А) именем ассоциации А–В
- Б) примечанием к ассоциации А–В
- В) классом ассоциации А–В



На рисунке

- А) А является обобщением В
- Б) А является композицией В



В) модель недопустима

На рисунке

- А) А использует интерфейс С
- Б) А использует интерфейс D
- В) В использует интерфейс С
- Г) В использует интерфейс D
- Д) модель недопустима

На рисунке

- А) А использует интерфейс В
- Б) А реализует интерфейс В
- В) С использует интерфейс В
- Г) С реализует интерфейс В
- Д) модель недопустима

На рисунке С является

- А) именем роли
- Б) именем связи
- В) именем сообщения
- Г) именем ассоциации
- Д) модель недопустима

На рисунке

- А) А является именем объекта
- Б) В является именем объекта
- В) А является именем класса
- Г) В является именем класса
- Д) модель недопустима

Отношение обобщения невозможно между

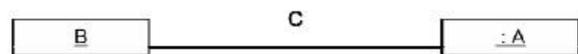
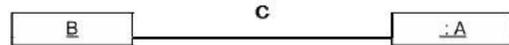
- А) классами
- Б) узлами
- В) состояниями
- Г) вариантами использования
- Д) действующими лицами

Отношение ассоциации невозможно между

- А) классами
- Б) узлами
- В) пакетами
- Г) вариантами использования
- Д) компонентами

Отношение зависимости невозможно между

- А) классами



- Б) узлами
- В) пакетами
- Г) вариантами использования
- Д) примечаниями

На канонических диаграммах классов не используются

- А) объекты
- Б) классы
- В) компоненты
- Г) узлы
- Д) интерфейсы

На канонических диаграммах объектов используются

- А) объекты
- Б) классы
- В) компоненты
- Г) узлы
- Д) интерфейсы

На канонических диаграммах компонентов не используются

- А) объекты
- Б) классы
- В) компоненты
- Г) узлы
- Д) интерфейсы

На канонических диаграммах размещения не используются

- А) объекты
- Б) классы
- В) компоненты
- Г) узлы
- Д) интерфейсы

На диаграмме взаимодействия UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

На диаграмме состояний UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы

- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

На диаграмме деятельности UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Действия

На диаграмме последовательности UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

На диаграмме кооперации (коммуникации) UML применяют следующие основные типы сущностей

- А) Классы
- Б) Варианты использования
- В) Действующие лица
- Г) Интерфейсы
- Д) Узлы
- Е) Состояния
- Ж) Объекты
- З) Компоненты

Графический комментарий «плавательные дорожки» применяется на

- А) Диаграммах классов
- Б) Диаграммах использования
- В) Диаграммах состояний
- Г) Диаграммах деятельности

На рисунке следующие сообщения являются недопустимыми на канонической диаграмме последовательности

- А) M1
- Б) M2
- В) M3
- Г) M4
- Д) M5
- Е) M6
- Ж) все допустимы

На рисунке объект А

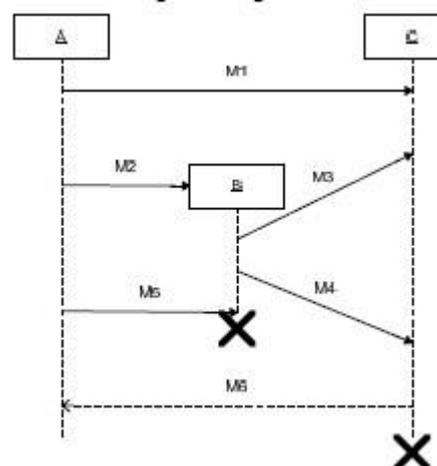
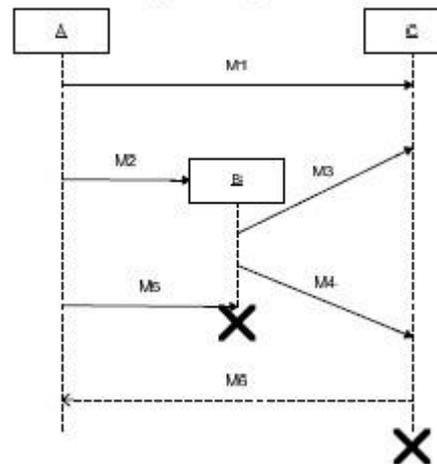
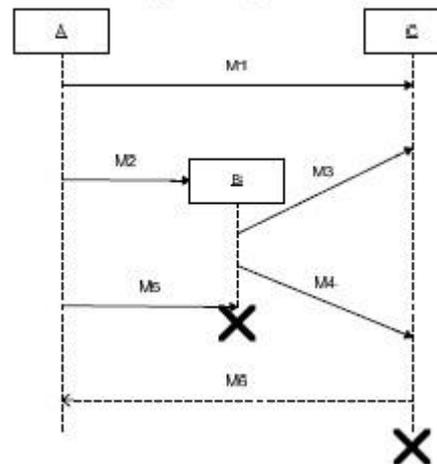
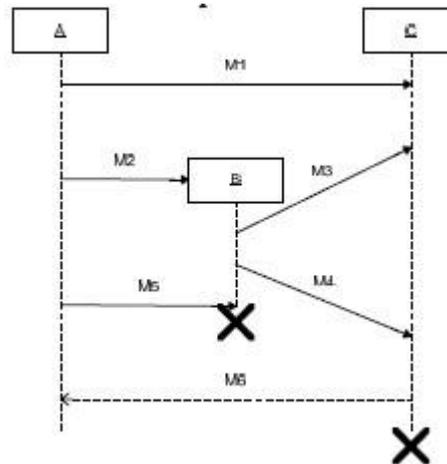
- А) существует до начала взаимодействия
- Б) возникает в процессе взаимодействия
- В) существует после окончания взаимодействия
- Г) уничтожается в процессе взаимодействия

На рисунке объект В

- А) существует до начала взаимодействия
- Б) возникает в процессе взаимодействия
- В) существует после окончания взаимодействия
- Г) уничтожается в процессе взаимодействия

На рисунке объект С

- А) существует до начала взаимодействия
- Б) возникает в процессе взаимодействия
- В) существует после окончания взаимодействия
- Г) уничтожается в процессе взаимодействия



Если все простые состояния достижимы, и машина состояний завершает свою работу, то любое простое состояние обязательно имеет

- А) Хотя бы один входящий переход
- Б) Хотя бы один исходящий переход
- В) Не более одного входящего перехода
- Г) Не более одного исходящего перехода

Простое состояние не может иметь

- А) Двух или более входящих переходов по одному событию
- Б) Двух или более исходящих переходов по одному событию
- В) Двух или более входящих спонтанных переходов без сторожевого условия
- Г) Двух или более исходящих спонтанных переходов без сторожевого условия

Простой (несегментированный) спонтанный переход обязательно имеет

- А) Ровно одно исходное состояние
- Б) Ровно одно целевое состояние
- В) Ровно одно событие перехода
- Г) Ровно одно сторожевое условие
- Д) Ровно одно действие на переходе

Сегментированный спонтанный переход может иметь

- А) несколько начальных состояний
- Б) несколько конечных состояний
- В) несколько событий перехода
- Г) несколько сторожевых условий
- Д) несколько действий на переходе

Простой (несегментированный) переход по событию может иметь

- А) несколько исходных состояний
- Б) несколько целевых состояний
- В) несколько событий перехода
- Г) несколько сторожевых условий
- Д) несколько действий на переходе

Простой (несегментированный) переход по событию обязательно имеет

- А) ровно одно исходное состояние
- Б) ровно одно целевое состояние
- В) ровно одно событие перехода
- Г) ровно одно сторожевое условие
- Д) ровно одно действие на переходе

Модель считается противоречивой, если множество исходящих переходов простого состояния

- А) пусто
- Б) содержит переходы с разными событиями и совместными сторожевыми условиями
- В) содержит переходы с одинаковыми событиями и совместными сторожевыми условиями
- Г) содержит переходы с разными событиями и несовместными сторожевыми условиями

Д) содержит переходы с одинаковыми событиями и несовместными сторожевыми условиями **Простое состояние не может содержать**

- А) действие при входе
- Б) действие при выходе
- В) сторожевое условие
- Г) внутренний переход

Модель считается противоречивой, если для данного события выполненными сторожевыми условиями

- А) пусто
- Б) содержит один элемент
- В) содержит более одного элемента

Считается, что сторожевое условие else выполнено, если

- А) все сторожевые условия на исходящих переходах для данного события из данного состояния выполнены
- Б) все сторожевые условия на исходящих переходах для данного события из данного состояния не выполнены
- В) хотя бы одно сторожевое условие на исходящих переходах для данного события из данного состояния выполнено
- Г) хотя бы одно сторожевое условие на исходящих переходах для данного события из данного состояния не выполнено

При изображении сегментированного перехода на диаграмме состояний сегменты можно разделять следующими значками (укажите лишнее)

- А) Круг
- Б) Ромб
- В) Флажок
- Г) Полоска

Корневой сегмент сегментированного перехода может содержать

- А) событие
- Б) сторожевое условие
- В) действие на переходе
- Г) сообщение

Листовой сегмент сегментированного перехода может содержать

- А) событие
- Б) сторожевое условие
- В) действие на переходе
- Г) сообщение

Сторожевые условия вдоль пути в сегментированном переходе соединены операцией

- А) ИЛИ
- Б) Исключающее ИЛИ
- В) И
- Г) НЕ

В UML используются следующие типы событий (укажите лишнее)

- А) событие вызова
- Б) событие сигнала
- В) событие таймера
- Г) событие создания
- Д) событие изменения

Событие таймера

- А) возникает в заданный момент времени
- Б) возникает по истечении заданного интервала времени
- В) возникает периодически с заданным периодом

Событие изменения

- А) возникает в заданный момент времени
- Б) возникает при любом изменении значения заданного логического выражения
- В) возникает при определенном изменении значения заданного логического выражения

Действие в UML может быть одного из следующих типов (укажите лишнее)

- А) присваивание значения
- Б) вызов операции
- В) создание объекта
- Г) уничтожение объекта
- Д) возврат значения
- Е) посылка сигнала
- Ж) прием сигнала

Если некоторая активность может быть прервана событием и может продолжаться неограниченно долго, то такая активность

- А) называется в UML действием
- Б) называется в UML деятельностью
- В) не определяется и не используется в UML

Если некоторая активность не может быть прервана событием и может продолжаться неограниченно долго, то такая активность

- А) называется в UML действием
- Б) называется в UML деятельностью
- В) не определяется и не используется в UML

Если некоторая активность не может быть прервана событием и не может продолжаться неограниченно долго, то такая активность

- А) называется в UML действием
- Б) называется в UML деятельностью
- В) не определяется и не используется в UML

На рисунке

- А) поведение не определено
- Б) выполнятся деятельности А или В, а затем С
- В) выполнятся деятельности А и В, а затем С
- Г) выполнение не закончится **На рисунке**

- А) поведение не определено
- Б) выполнятся деятельности А или В,а затем С
- В) выполнятся деятельности А и В, а затем С
- Г) выполнение не закончится

На рисунке

- А) поведение не определено
- Б) выполнятся деятельности А или В, а затем С
- В) выполнятся деятельности А и В, а затем С
- Г) выполнение не закончится

На рисунке

- А) поведение не определено
- Б) выполнятся деятельности А или В, а затем С
- В) выполнятся деятельности А и В, а затем С
- Г) выполнение не закончится

На канонических диаграммах деятельности не используются

- А) объекты
- Б) дорожки
- В) сообщения
- Г) переходы

На канонических диаграммах последовательности не используются

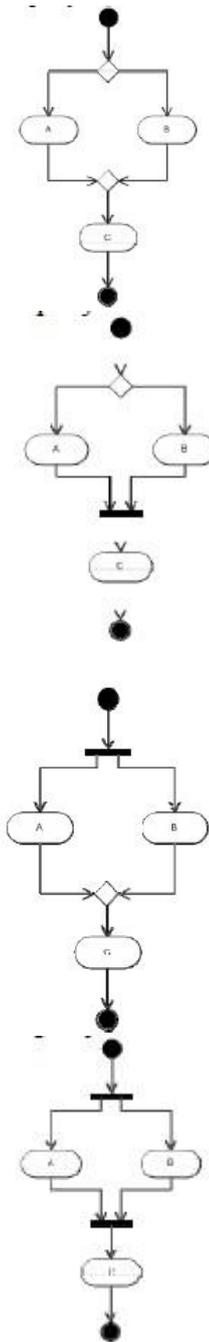
- А) объекты
- Б) дорожки
- В) сообщения
- Г) переходы

На канонических диаграммах кооперации не используются

- А) объекты
- Б) дорожки
- В) сообщения
- Г) переходы

На канонических диаграммах состояний не используются

- А) объекты
- Б) дорожки



- В)сообщя
- Г)перехоы

Графическая нотация UML 2 использует графический элемент, который не используется в UML 1.x

- А) Рамка
- Б) Фигура
- В) Значок
- Г)Лния
- Д)Текст

В графической нотации UML при изображении фигур имеет значение

- А) Форма
- Б) Размер
- В) Соотношение геометрических параметров (например, отношение длины к ширине)
- Г) Взаимное положение фигур

В графической нотации UML при изображении линий имеет значение

- А) Толщина
- Б) Цвет
- В) Тип (прямая, ломаная, кривая)

Г) Стиль (сплошная, пунктирная, волнистая) В графической нотации UML при изображении текстов имеет значение

- А) Гарнитура шрифта (Times New Roman ...)
- Б) Начертание шрифта (полужирный ...)
- В) Цвет шрифта (черный ...)
- Г) Кегль шрифта (10 пунктов ...)

Система программирования обязательно содержит

- А) Транслятор языка программирования
- Б) Графический редактор диаграмм
- В) Символьный отладчик

Интегрированная среда разработки обязательно содержит

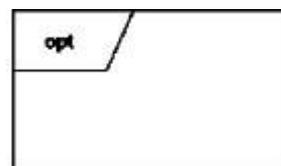
- А) Транслятор языка программирования
- Б) Графический редактор диаграмм
- В) Символьный отладчик

CASE-средство обязательно содержит

- А) Транслятор языка программирования
- Б) Графический редактор диаграмм
- В) Символьный отладчик

Приведенная на рисунке фигура является

- А) Классом
- Б) Пакетом
- В) Объектом
- Г) Составным шагом
- Д) Рамкой
- Е) Примечанием



Ж) Не используется в UML

Приведенная на рисунке фигура Д) Рамкой

является

Е) Примечанием

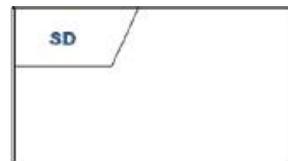
А) Классом

Ж) Не используется в UML

Б) Пакетом

В) Объектом

Г) Составным шагом



Приведенная на рисунке фигура является А) Классом

Б) Пакетом

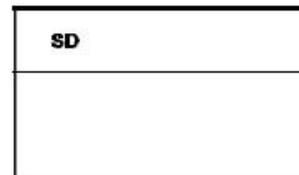
В) Объектом

Г) Составным шагом

Д) Рамкой

Е) Примечанием

Ж) Не используется в UML



Приведенная на рисунке фигура является

А) Классом

Б) Пакетом

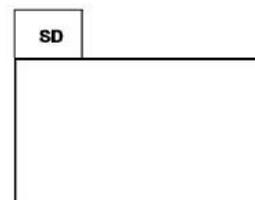
В) Объектом

Г) Составным шагом

Д) Рамкой

Е) Примечанием

Ж) Не используется в UML



Критерии оценки:

90-100 баллов **«отлично»** заслуживает обучающийся, показавший всестороннее систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой; как правило, оценка «отлично» выставляется обучающимся, усвоившим взаимосвязь основных понятий междисциплинарного курса и их значение для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

80-90 баллов **«хорошо»** заслуживает обучающийся, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе; как правило, оценка «хорошо» выставляется обучающимся, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;

60-80 баллов **«удовлетворительно»** заслуживает обучающийся, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой; как правило, оценка «удовлетворительно» выставляется обучающимся, допустившим погрешности в ответе на

зачете, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

Менее 60 баллов «неудовлетворительно» выставляется обучающемуся, обнаружившему проблемы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий; как правило, оценка «неудовлетворительно» ставится обучающимся, которые не могут продолжать обучение или приступить к профессиональной деятельности по окончании учебного заведения без дополнительных занятий по соответствующему междисциплинарному курсу.

Примерная тематика курсовых работ

1. Построение архитектуры программного средства
2. Диаграммы переходов состояний ПО
3. Функциональные диаграммы ПО
4. Метод пошаговой детализации алгоритма и программы
5. Проектирование структур данных ПО
6. Оценка программных средств с помощью метрик
7. Инспекция программного кода на предмет соответствия стандартам кодирования
8. Предотвращение, обнаружение и исправление ошибок в разработке ПО
9. Общие требования к графическому интерфейсу пользователя
10. Современные принципы и методы разработки программных приложений
11. Основные подходы к интегрированию программных модулей
12. Стандарты кодирования ПО и их применение
13. Модели жизненного цикла разработки программного продукта
14. Понятие жизненного цикла программного продукта
15. Взаимосвязь между процессами жизненного цикла программного продукта
16. Цели и задачи и виды тестирования программ
17. Меры и метрики. Тестовое покрытие
18. Стандарты, регламентирующие процесс разработки программного обеспечения
19. Документация процесса разработки ПО
20. Документация по сопровождению ПО. Стандарты документации ПО
21. Верификация и аттестация программного обеспечения
22. Эволюция моделей жизненного цикла программного обеспечения
23. Алгоритмизация и программирование на языках высокого уровня
24. Методы тестирования информационных систем
25. Проектирование и разработка ПО автоматизированной информационной системы «Материальный склад»
26. Разработка имитационной модели программного обеспечения информационной системы "Центр обслуживания абонентов"
27. Паскаль как язык структурного ориентированного программирования
28. Продвижение и сопровождение программного обеспечения в пассажирских перевозках
29. Продвижение и сопровождение программного обеспечения в предприятии малого бизнеса
30. Разработка и внедрение модуля интеграции 1С
31. Разработка информационной системы для туристического агентства
32. Разработка информационной системы для интернет-магазина
33. Разработка информационной системы по продаже автомобилей
34. Разработка модели программного обеспечения информационной системы автоматизации работы рекламного агентства
35. Разработка программного обеспечения информационной системы «Ювелирный магазин»
36. Разработка модуля информационной системы для управления капитальным ремонтом
37. Разработка программного модуля бизнес-процессов
38. Разработка программных модулей программного обеспечения для компьютерных систем
39. Диаграммы переходов состояний в языках программирования

40. Модульное проектирование Программной системы
41. Описание информационной системы при помощи диаграмм UML
42. Исследование и описание основных характеристик и требований к разработке информационной системы для магазина компьютерной техники.
43. Функциональное моделирование деятельности информационной системы
44. Моделирование сценариев взаимодействия с помощью диаграмм взаимодействия и диаграмм состояния.
45. Разработки диаграмм состояния на основе вариантов использования. Примеры.
46. Понятие технологии разработки программного обеспечения.
47. Средства автоматизации разработки экспертных систем
48. Средства тестирования программ
49. CASE-средства и визуальное моделирование
50. Проектирование информационной системы с использованием PlantUML

Критерии оценки:

Оценка «5»: ответ полный и правильный на основании изученных теорий; материал изложен в определенной логической последовательности, литературным языком. Ответ самостоятельный.

Оценка «4»: ответ полный и правильный на основании изученных теорий; материал изложен в определенной логической последовательности, при этом допущены две-три несущественные ошибки, исправленные по требованию преподавателя.

Оценка «3»: ответ полный, но при этом допущена существенная ошибка, или неполный, несвязный.

Оценка «2»: при ответе обнаружено непонимание обучающимся основного содержания учебного материала или допущены существенные ошибки, которые он не смог исправить при наводящих вопросах преподавателя.

Комплект лабораторно-практических заданий

по профессиональному модулю

«Осуществление интеграции программных модулей»

Практическое занятие «Анализ предметной области»

1. Практическое занятие «Разработка и оформление технического задания»
2. Практическое занятие «Построение архитектуры программного средства»
3. Практическое занятие «Изучение работы в системе контроля версий»
4. Лабораторная работа «Построение диаграммы Вариантов использования
5. и диаграммы. Последовательности»
6. Лабораторная работа «Построение диаграммы Кооперации и диаграммы Развертывания»
7. Лабораторная работа «Построение диаграммы Деятельности, диаграммы Состояний и диаграммы Классов»
8. Лабораторная работа «Построение диаграммы компонентов»
9. Лабораторная работа «Построение диаграмм потоков данных»
10. Лабораторная работа «Разработка тестового сценария»

11. Лабораторная работа «Оценка необходимого количества тестов»
12. Лабораторные работы «Разработка тестовых пакетов»
13. Лабораторные работы «Оценка программных средств с помощью метрик»
14. Лабораторные работы «Инспекция программного кода на предмет соответствия стандартам кодирования»
15. Лабораторная работа «Разработка структуры проекта»
16. Лабораторная работа «Разработка модульной структуры проекта (диаграммы модулей)»
17. Лабораторная работа «Разработка перечня артефактов и протоколов проекта»
18. Лабораторная работа «Настройка работы системы контроля версий (типов импортируемых файлов, путей, фильтров и др. параметров импорта в репозиторий)»
19. Лабораторная работа «Разработка и интеграция модулей проекта (командная работа)»
20. Лабораторная работа «Отладка отдельных модулей программного проекта»
21. Лабораторная работа «Организация обработки исключений»
22. Лабораторная работа «Применение отладочных классов в проекте»
23. Лабораторная работа «Отладка проекта»
24. Лабораторная работа «Инспекция кода модулей проекта»
25. Лабораторная работа «Тестирование интерфейса пользователя средствами инструментальной среды разработки»
26. Лабораторная работа «Разработка тестовых модулей проекта для тестирования отдельных модулей»
27. Лабораторная работа «Выполнение функционального тестирования»
28. Лабораторная работа «Тестирование интеграции»
29. Лабораторная работа «Документирование результатов тестирования»
30. Лабораторная работа «Построение простейших математических моделей. Построение простейших статистических моделей»
31. Лабораторная работа «Решение простейших однокритериальных задач»
32. Лабораторная работа «Задача Коши для уравнения теплопроводности»
33. Лабораторная работа «Сведение произвольной задачи линейного программирования к основной задаче линейного программирования»
34. Лабораторная работа «Решение задач линейного программирования симплекс–методом»
35. Лабораторная работа «Нахождение начального решения транспортной задачи. Решение транспортной задачи методом потенциалов»
36. Лабораторная работа «Применение метода стрельбы для решения линейной краевой задачи»
37. Лабораторная работа «Задача о распределении средств между предприятиями»
38. Лабораторная работа «Задача о замене оборудования»
39. Лабораторная работа «Составление систем уравнений Колмогорова. Нахождение финальных вероятностей. Нахождение характеристик простейших систем массового обслуживания»
40. Лабораторная работа «Решение задач массового обслуживания методами имитационного моделирования»

41. Лабораторная работа «Построение прогнозов»
42. Лабораторная работа «Решение матричной игры методом итераций»
43. Лабораторная работа «Моделирование прогноза»

Учебная практика по модулю

Виды работ:

1. Вводный инструктаж по технике безопасности во время прохождения практики.
2. Разработка технического задания.
3. Выработка требований к программному обеспечению и программному модулю.
4. Проектирование программного обеспечения для решения прикладных задач.
5. Построение структуры программного продукта.
6. Кодирование программного обеспечения.
7. Тестирование и сопровождение программного обеспечения.
8. Проведение функционального и оценочного тестирования готового программного продукта.
9. Разработка и оформление технической документации.
10. Составление описания на программный продукт.
11. Администрирование программного обеспечения.
12. Подготовка презентаций для защиты программных продуктов.

Производственная практика по модулю

Виды работ

1. Вводный инструктаж по технике безопасности
 2. Изучение должностной характеристик предприятия как объекта компьютеризации
 3. Изучение инструментальных средств разработки программного обеспечения предприятия
 4. Проектирование программных систем для решения прикладных задач организации
 - Определение требований к программному продукту
 5. Составление технического задания (ТЗ)
 6. Выбор возможных способов реализации ТЗ
 7. Описание предметной области
 8. Выбор программных и аппаратных средств реализации программного продукта
 9. Проектирование программного продукта
 10. Разработка диаграммы вариантов
 11. Разработка диаграммы классов
 12. Разработка диаграммы последовательности
 13. Разработка диаграммы компонентов
 14. Разработка структурной схемы программного продукта
 15. Проектирование модулей программного продукта
 16. Разработка и оформление требований к программным модулям по предложенной документации
 17. Разработка алгоритм программного продукта
 18. Разработка программного кода
 19. Реализация программного кода с использованием инструментальных программных средств
 20. Разработка интерфейса программного продукта
 21. Отладка программного продукта
 22. Тестирование программного продукта
 23. Инспектирование разработанных программных модулей на предмет соответствия стандартам кодирования
 24. Реализация интерфейса пользователя
 25. Составление сопроводительной документации на программный продукт
- Использование специализированных графических средства построения и анализа архитектуры программных продукт

Перечень вопросов для квалификационного экзамена

по профессиональному модулю

«Осуществление интеграции программных модулей»

1. Понятия требований, классификация, уровни требований. Методологии и стандарты, регламентирующие работу с требованиями.
2. Современные принципы и методы разработки программных приложений.
3. Методы организации работы в команде разработчиков. Системы контроля версий
4. Основные подходы к интегрированию программных модулей.
5. Стандарты кодирования.
6. Описание требований: унифицированный язык моделирования. Диаграммы UML.
7. Описание и оформление требований (спецификация). Анализ требований и стратегии выбора решения
8. Цели и задачи и виды тестирования. Стандарты качества программной документации. Меры и метрики.
9. Тестовое покрытие.
10. Тестовый сценарий, тестовый пакет.
11. Анализ спецификаций. Верификация и аттестация программного обеспечения.
12. Понятие репозитория проекта, структура проекта.
13. Виды, цели и уровни интеграции программных модулей. Автоматизация бизнес-процессов.
14. Выбор источников и приемников данных, сопоставление объектов данных.
15. Транспортные протоколы. Стандарты форматирования сообщений.
16. Организация работы команды в системе контроля версий.
17. Отладка программных продуктов. Инструменты отладки. Отладочные классы.
18. Ручное и автоматизированное тестирование.
19. Методы и средства организации тестирования.
20. Инструментарии анализа качества программных продуктов в среде разработке.
21. Обработка исключительных ситуаций.
22. Методы и способы идентификации сбоев и ошибок.
23. Выявление ошибок системных компонентов.
24. Понятие решения. Множество решений, оптимальное решение.
25. Показатель эффективности решения
26. Математические модели, принципы их построения, виды моделей.
27. Задачи: классификация, методы решения, граничные условия.
28. Общий вид и основная задача линейного программирования.
29. Симплекс – метод.
30. Транспортная задача. Методы нахождения начального решения транспортной задачи.
31. Метод потенциалов.
32. Общий вид задач нелинейного программирования.
33. Графический метод решения задач нелинейного программирования.
34. Метод множителей Лагранжа.

35. Основные понятия динамического программирования: шаговое управление, управление операцией в целом, оптимальное управление, выигрыш на данном шаге, выигрыш за всю операцию, аддитивный критерий, мультипликативный критерий.
36. Простейшие задачи, решаемые методом динамического программирования.
37. Методы хранения графов в памяти ЭВМ. Задача о нахождении кратчайших путей в графе и методы ее решения.
38. Задача о максимальном потоке и алгоритм Форда–Фалкерсона.
39. Системы массового обслуживания: понятия, примеры, модели.
40. Основные понятия теории марковских процессов: случайный процесс, марковский процесс, граф состояний, поток событий, вероятность состояния, уравнения Колмогорова, финальные вероятности состояний.
41. Схема гибели и размножения.
42. Метод имитационного моделирования. Единичный жребий и формы его организации. Примеры задач
43. Понятие прогноза. Количественные методы прогнозирования: скользящие средние, экспоненциальное сглаживание, проектирование тренда.
44. Качественные методы прогноза
45. Предмет и задачи теории игр. Основные понятия теории игр: игра, игроки, партия, выигрыш, проигрыш, ход, личные и случайные ходы, стратегические игры, стратегия, оптимальная стратегия.
46. Антагонистические матричные игры: чистые и смешанные стратегии.
47. Методы решения конечных игр: сведение игры $n \times n$ к задаче линейного программирования, численный метод – метод итераций.
48. Область применимости теории принятия решений. Принятие решений в условиях определенности, в условиях риска, в условиях неопределенности.
49. Критерии принятия решений в условиях неопределенности.
50. Дерево решений.

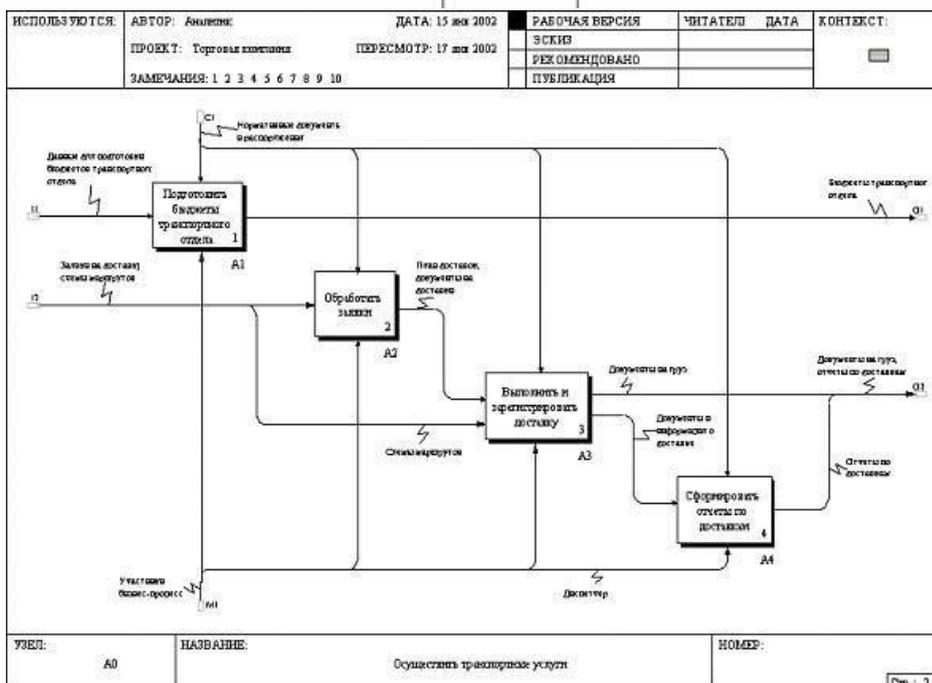
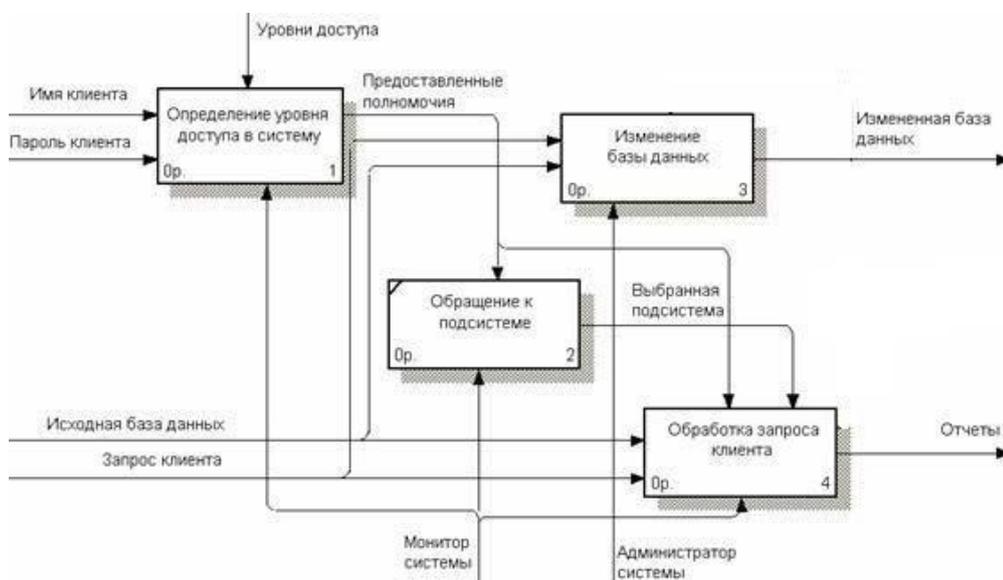
Перечень практических заданий для квалификационного экзамена по профессиональному модулю

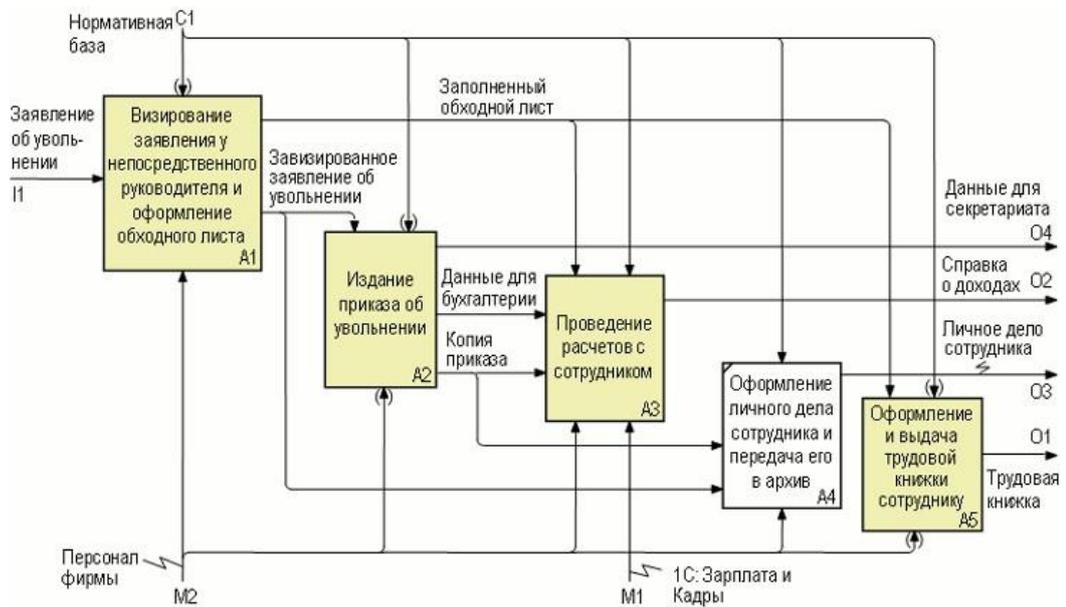
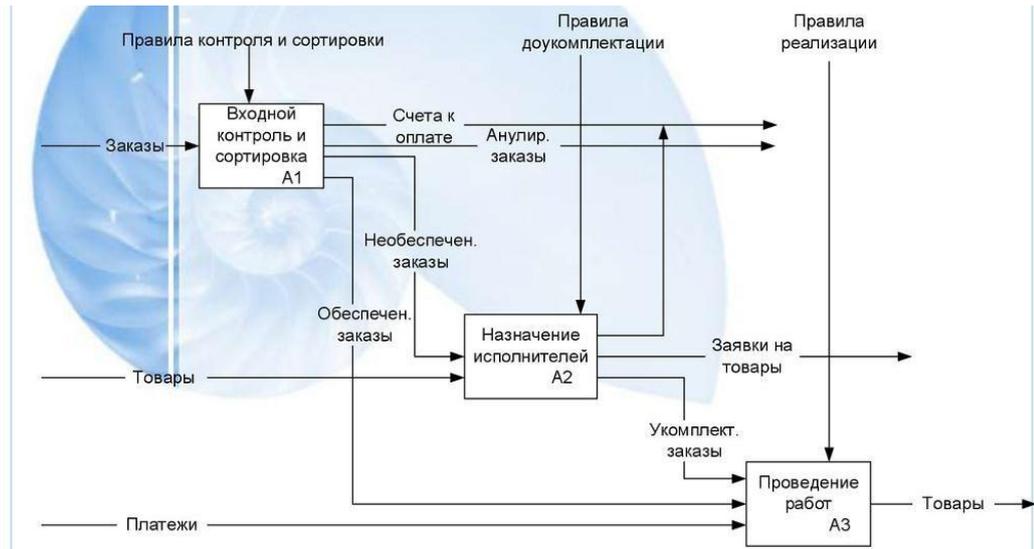
«Осуществление интеграции программных модулей»

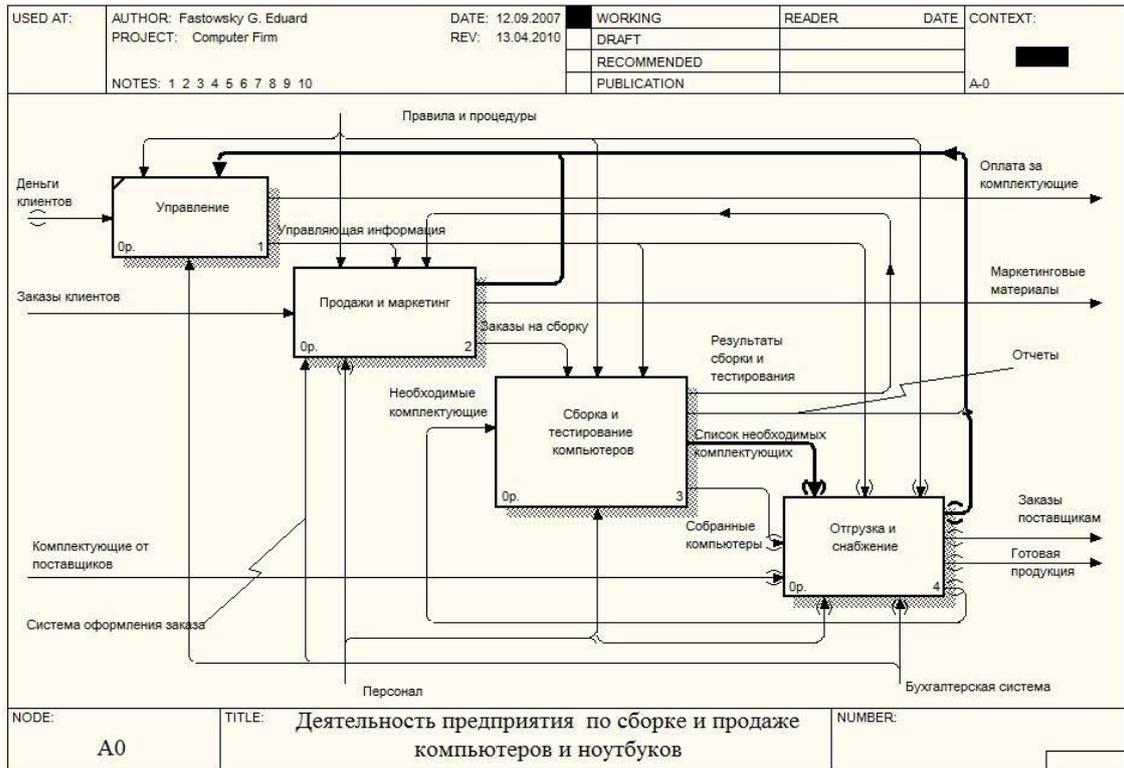
ПРАКТИЧЕСКОЕ ЗАДАНИЕ (в каждом задании)

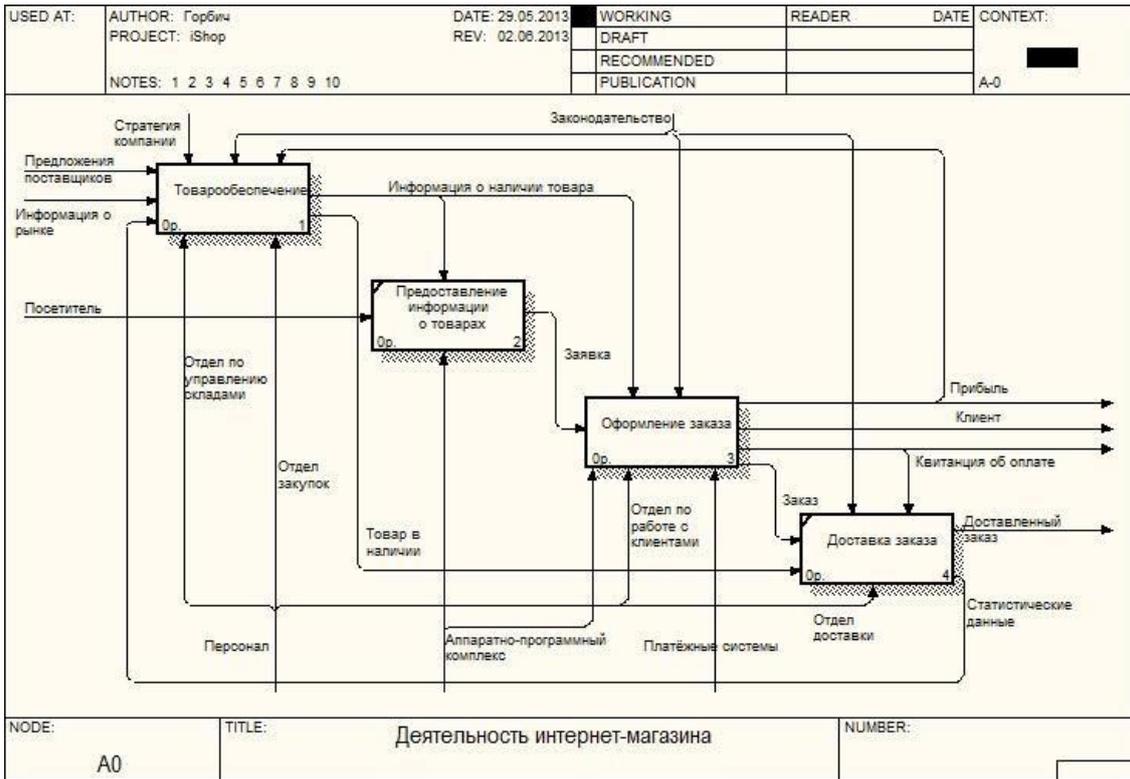
- A. Описание готовой функциональной диаграммы
- B. Описание готовой диаграммы классов
- C. Разработка программного продукта в Visual Studio

Функциональные диаграммы





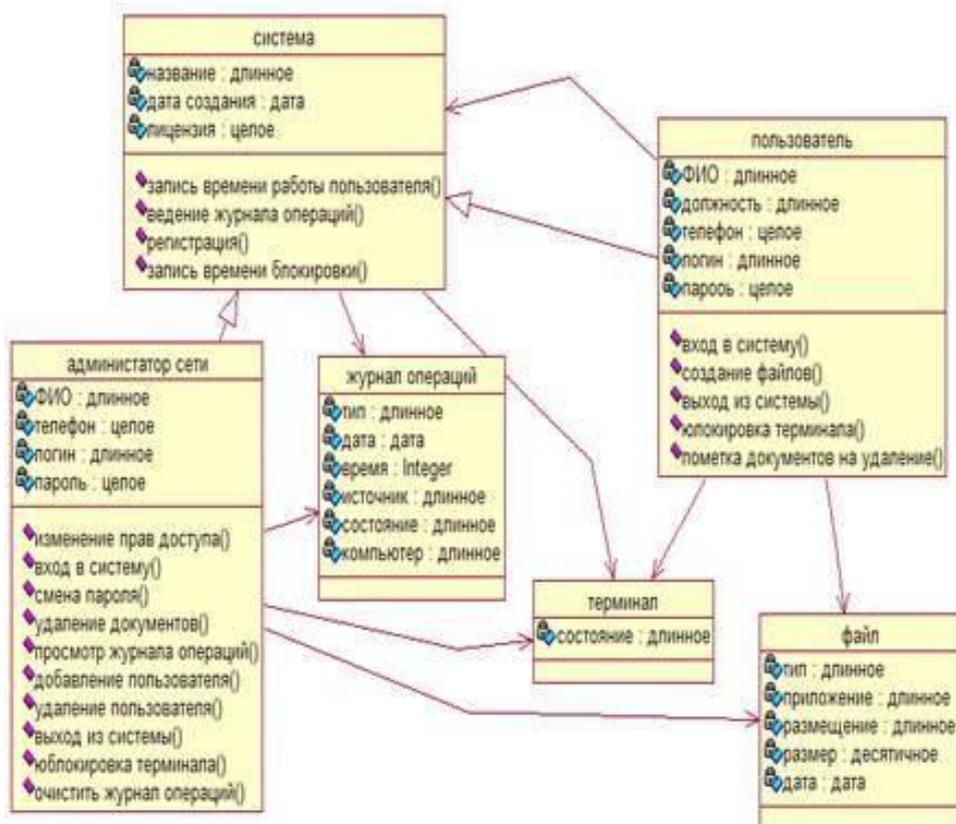
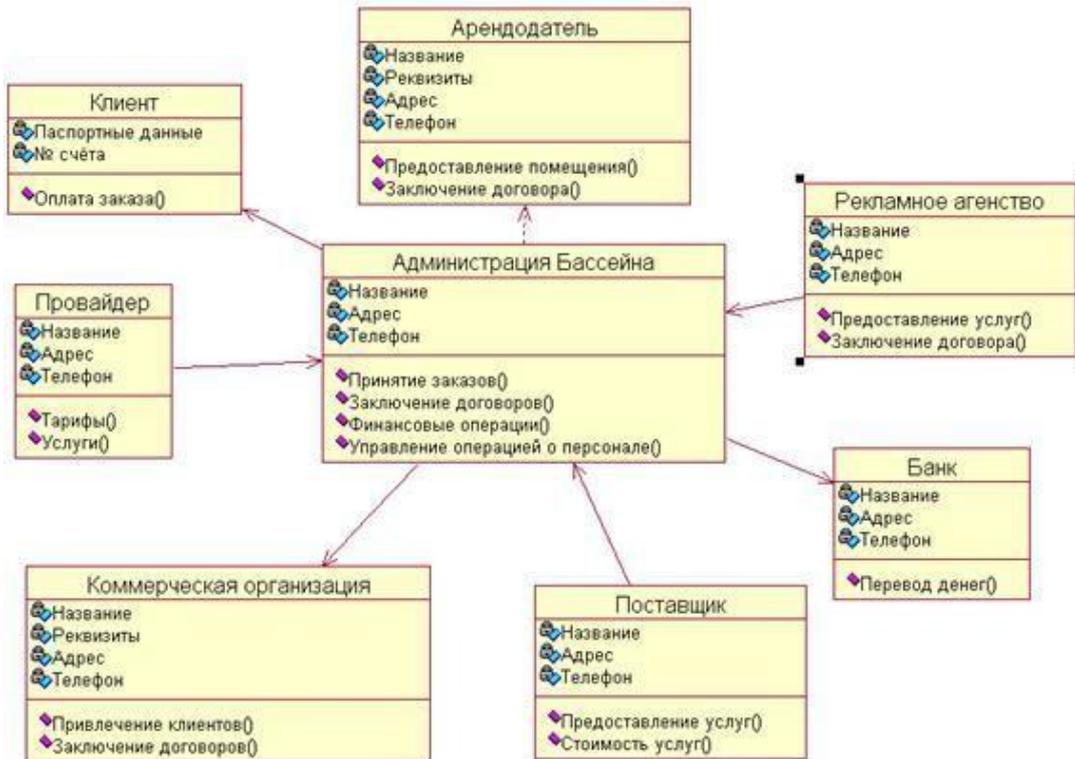


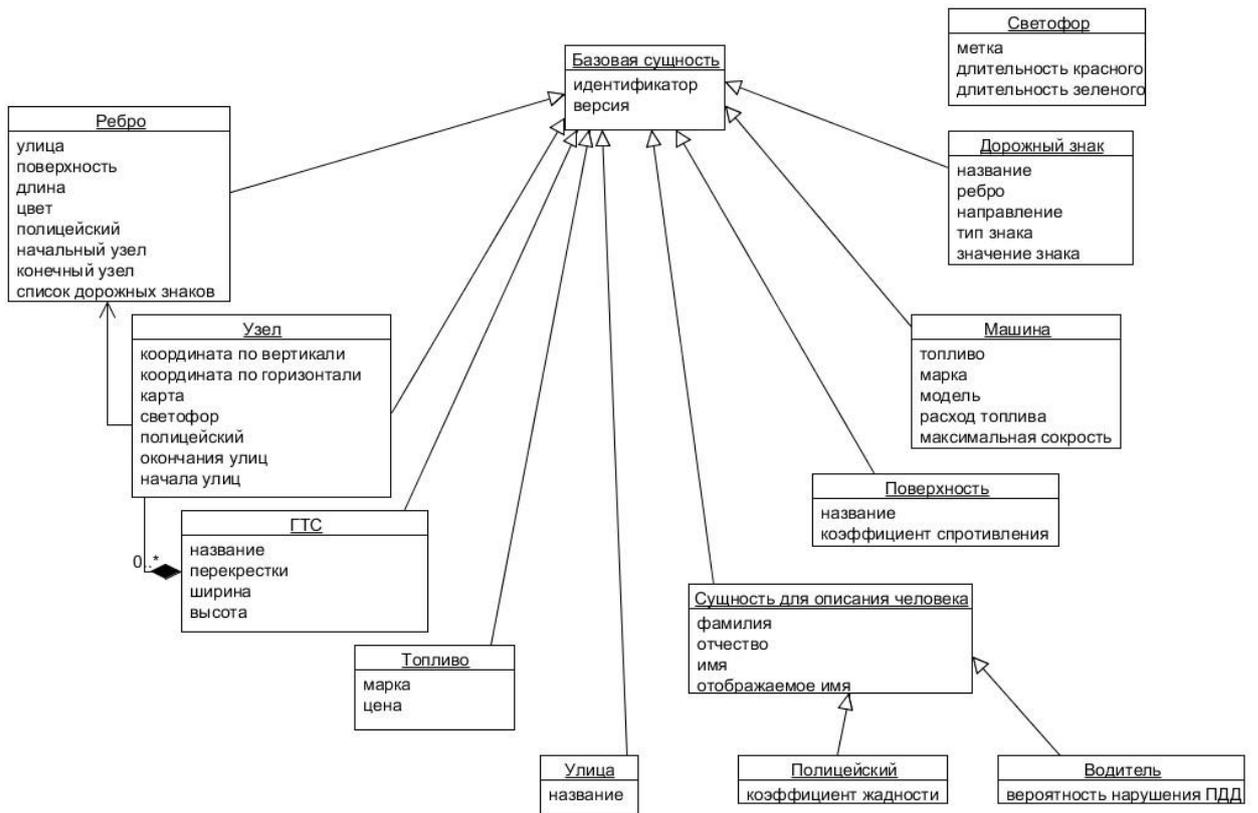
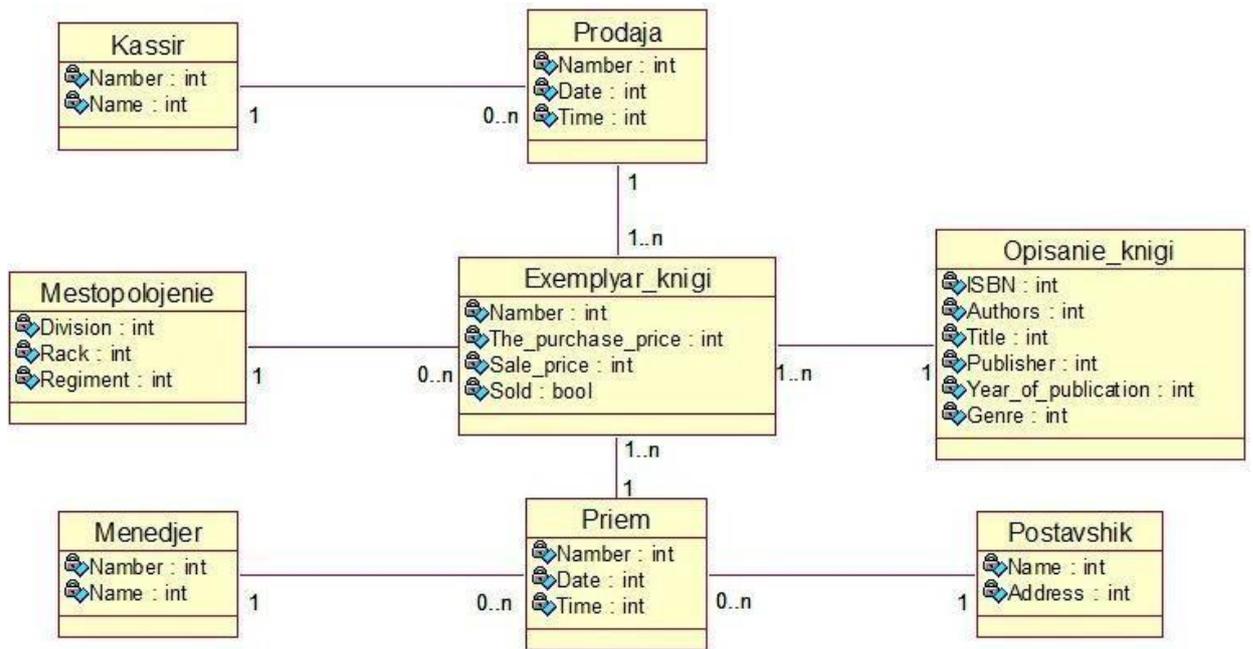


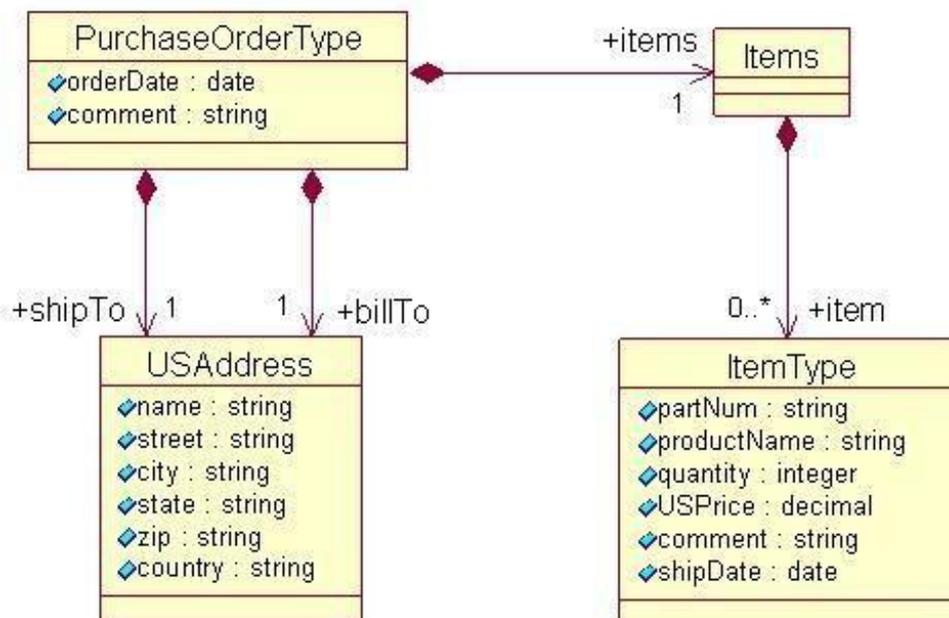
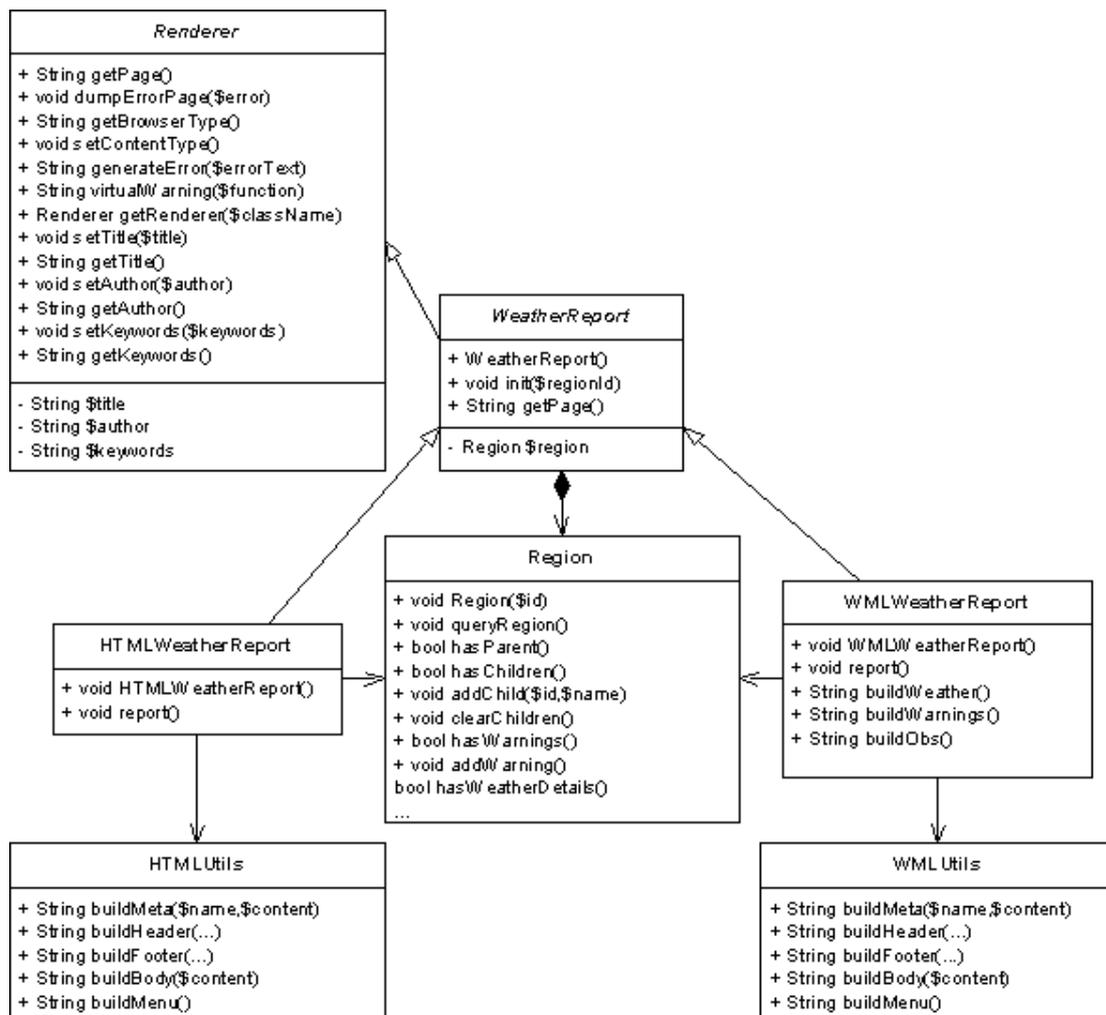
NODE: A0 TITLE: Деятельность интернет-магазина NUMBER:

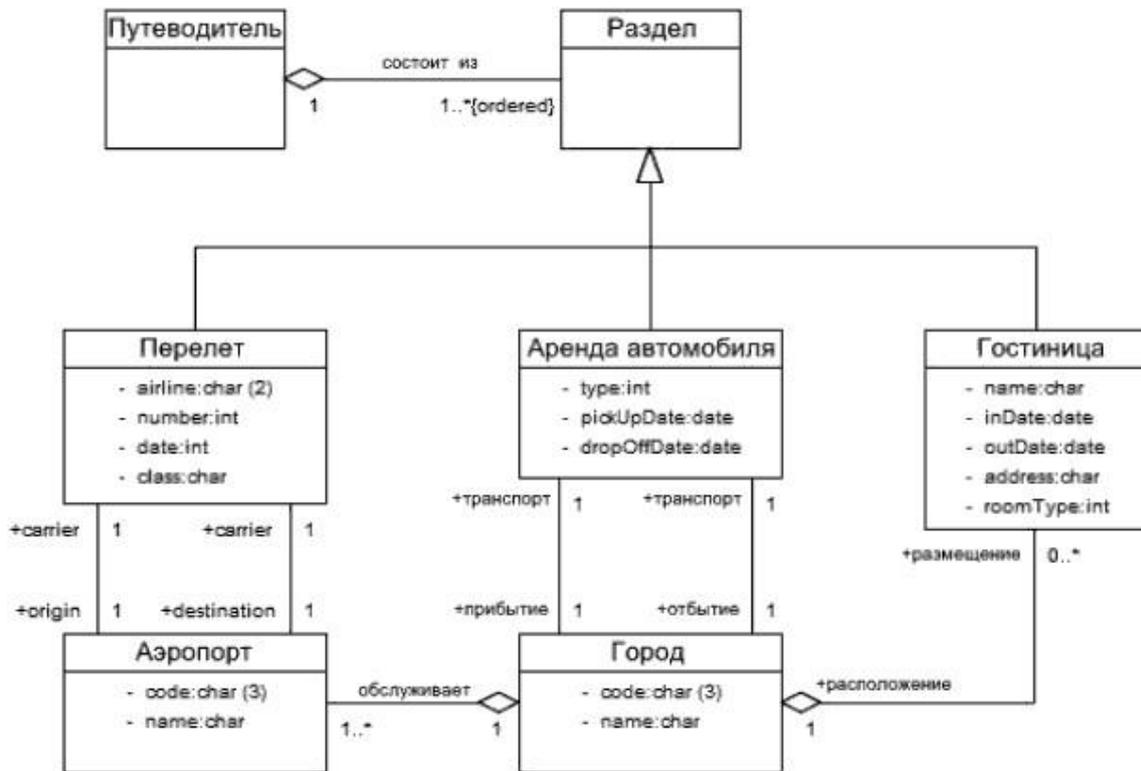


Диаграммы классов









Министерство образования и науки Республики Дагестан
Государственное бюджетное профессиональное образовательное учреждение РД
«Колледж экономики и права»

Специальность 09.02.07«Информационные системы и программирование»
Курс – 3, семестр – 6
Форма обучения очная

Экзаменационный БИЛЕТ № _____

по ПМ.02. Осуществление интеграции программных модулей

1. Теорет. вопр.-1
2. Теорет. вопр.-2
3. Практич. задание

Составитель _____ / _____ /

Утвержден на заседании цикловой комиссии _____ (протокол № __)

Председатель ПЦК _____

КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ПРОХОЖДЕНИЯ УЧЕБНОЙ ПРАКТИКИ (ПО ПРОФИЛЮ СПЕЦИАЛЬНОСТИ) ПО ПМ.02 ОСУЩЕСТВЛЕНИЕ ИНТЕГРАЦИИ ПРОГРАММНЫХ МОДУЛЕЙ

(название профессионального модуля)

Текущий контроль результатов прохождения учебной практики (по профилю специальности) происходит при использовании следующих обязательных форм контроля:

- ежедневный контроль посещаемости практики;
- наблюдением за выполнением видов работ на практике, предусмотренных программой практики;
- контроль качества выполнения видов работ по практике (уровень овладения ПК и ОК при выполнении работ оценивается в аттестационном листе руководителя практики от предприятия);
- контроль за ведением дневника практики;
- контроль сбора материала для отчета по практике, в соответствии с заданием на практику.

Форма промежуточной аттестации по учебной практике (по профилю специальности) по ПМ.02 Осуществление интеграции программных модулей - **зачет**.

Практика завершается зачетом при условии:

- полноты и своевременности представления дневника прохождения производственной практики и отчета по производственной практике в соответствии с заданием на практику.

Виды работ и проверяемые результаты учебной практики (по профилю специальности)

ОК, ПК	Виды работ	Формы и методы контроля для оценки результатов обучения
<p><i>ОК 01.</i> <i>ОК 02.</i> <i>ОК 03.</i> <i>ОК 04.</i> <i>ОК 05.</i> <i>ОК 06.</i> <i>ОК 07.</i> <i>ОК 08.</i> <i>ОК 09.</i> <i>ПК 2.1.</i> <i>ПК 2.2.</i> <i>ПК 2.3.</i> <i>ПК 2.4.</i> <i>ПК 2.5.</i></p>	<p>Инструктаж по технике безопасности Стандартизация требований к программному обеспечению Описание и анализ требований. Диаграммы IDEF Оценка качества программных средств Современные технологии и инструменты интеграции Инструментарий тестирования и анализа качества программных средств Основы моделирования. Детерминированные задачи Задачи в условиях неопределенности Оформление дневника-отчета, индивидуального задания, презентации Подведение итогов практики</p>	<p>Полнота и своевременность представления дневника прохождения учебной практики и отчета по учебной практике в соответствии с заданием на практику.</p>

**КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ПРОХОЖДЕНИЯ
ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ (ПО ПРОФИЛЮ
СПЕЦИАЛЬНОСТИ)
ПО ПМ.02 ОСУЩЕСТВЛЕНИЕ ИНТЕГРАЦИИ ПРОГРАММНЫХ
МОДУЛЕЙ**

ОК, ПК	Виды работ	Формы и методы контроля для оценки результатов обучения
<p><i>ОК 01.</i> <i>ОК 02.</i> <i>ОК 03.</i> <i>ОК 04.</i> <i>ОК 05.</i> <i>ОК 06.</i> <i>ОК 07.</i> <i>ОК 08.</i> <i>ОК 09.</i> <i>ПК 2.1.</i> <i>ПК 2.2.</i> <i>ПК 2.3.</i> <i>ПК 2.4.</i> <i>ПК 2.5.</i></p>	<p>Инструктаж по технике безопасности Знакомство с предприятием (организацией) Изучение организации работы на предприятии (организации) Изучение структуры органов управления предприятием Анализ системного и прикладного программного обеспечения Анализ технических средств информации предприятия Анализ сетевого программного обеспечения предприятия Разработка требований к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент Интеграция модулей в программное обеспечение Отладка программного модуля с использованием специализированных программных средств Разработка тестовых наборов и тестовых сценариев для программного обеспечения Инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования Оформление дневника-отчета, индивидуального задания, презентации Подведение итогов практики</p>	<p>Наличие положительного аттестационного листа по практике руководителя практики - работника предприятия, закрепленного в качестве руководителя об уровне освоения общих и профессиональных компетенций в период прохождения практики. Наличие положительной производственной характеристики (отзыва) на обучающегося руководителя практики - работника предприятия, закрепленного в качестве руководителя; Полнота и своевременность представления дневника прохождения производственной практики и отчета по производственной практике в соответствии с заданием на практику.</p>

(название профессионального модуля)

Текущий контроль результатов прохождения производственной практики (по профилю специальности) происходит при использовании следующих обязательных форм контроля:

- ежедневный контроль посещаемости практики;
- наблюдением за выполнением видов работ на практике, предусмотренных программой практики;

- контроль качества выполнения видов работ по практике (уровень овладения ПК и ОК при выполнении работ оценивается в аттестационном листе руководителя практики от предприятия);
- контроль за ведением дневника практики;
- контроль сбора материала для отчета по практике, в соответствии с заданием на практику.

Форма промежуточной аттестации по производственной практике (по профилю специальности) по ПМ.02 Осуществление интеграции программных модулей - **зачет**.

Практика завершается зачетом при условии:

- положительного аттестационного листа по практике руководителя практики - работника предприятия, закрепленного в качестве руководителя об уровне освоения общих и профессиональных компетенций в период прохождения практики; - наличия положительной производственной характеристики (отзыва) на обучающегося руководителя практики - работника предприятия, закрепленного в качестве руководителя;
- полноты и своевременности представления дневника прохождения производственной практики и отчета по производственной практике в соответствии с заданием на практику.

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

СВЕДЕНИЯ О СЕРТИФИКАТЕ ЭП

Сертификат 133600552358087161194895262509558337786447861787

Владелец Гайдаров Насир Алиевич

Действителен с 22.03.2024 по 22.03.2025